Theoretical Computer Science 622 (2016) 1–15

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

The k-distinct language: Parameterized automata constructions ☆

Ran Ben-Basat *, Ariel Gabizon, Meirav Zehavi

Department of Computer Science, Technion – Israel Institute of Technology, Haifa 32000, Israel

ARTICLE INFO

Article history:
Received 28 October 2014
Received in revised form 28 December 2015
Accepted 25 January 2016
Available online 2 February 2016
Communicated by Z. Esik

Keywords:
Parameterized complexity
Automata theory
k-Path
r-Dimensional k-matching

ABSTRACT

In this paper, we pioneer a study of parameterized automata constructions for languages related to the design of parameterized algorithms. We focus on the k-Distinct language $L_k(\Sigma) \subseteq \Sigma^k$, defined as the set of words of length k over an alphabet Σ whose symbols are all distinct. This language is implicitly related to several breakthrough techniques developed during the last two decades, to design parameterized algorithms for fundamental problems such as k-Path and r-Dimensional k-Matching. Building upon the color coding, divide-and-color and narrow sieves techniques, we obtain the following automata constructions for $L_k(\Sigma)$. We develop non-deterministic automata (NFAs) of sizes $4^{k+o(k)} \cdot n^{O(1)}$ and $(2e)^{k+o(k)} \cdot n^{O(1)}$, where the latter satisfies a 'bounded ambiguity' property relevant to approximate counting, as well as a non-deterministic xor automaton (NXA) of size $2^k \cdot n^{O(1)}$, where $n = |\Sigma|$. We show that our constructions can be used to develop both deterministic and randomized algorithms for k-Path, r-Dimensional k-Matching and Module Motif in a natural manner, considering also their approximate counting variants. Our framework is modular and consists of two parts: designing an automaton for k-Distinct, and designing a problem specific automaton, as well as an algorithm for deciding whether the intersection automaton's language is empty, or for counting the number of accepting paths in it.

1. Introduction

Parameterized algorithms solve NP-hard problems by confining the combinatorial explosion to a parameter k. More precisely, a problem is *fixed-parameter tractable (FPT)* with respect to a parameter k if it can be solved in time $O^*(f(k))$ for some function f, where O^* hides factors polynomial in the input size.

In this paper, we pioneer the study of parameterized automata constructions for languages related to the design of parameterized algorithms. We focus on the k-Distinct language, consisting of all words of length k over an alphabet Σ, in which all of the symbols are distinct. Formally, let Σ be an alphabet of size n, and k a positive integer. The k-Distinct language, denoted $L_k(\Sigma) \subseteq \Sigma^k$, is the set of words $w_1 \cdots w_k \in \Sigma^k$, where $w_1, \dots, w_k$ are all distinct. To the best of our knowledge, this paper is the first to study k-Distinct and its applications.

☆ A preliminary version of this paper appeared in the proceedings of the 9th *International Symposium on Parameterized and Exact Computation (IPEC'14)* [1].
The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement number 257575.

* Corresponding author. Tel.: +972 4 8292449.
E-mail addresses: sran@cs.technion.ac.il (R. Ben-Basat), arielga@cs.technion.ac.il (A. Gabizon), meizeh@cs.technion.ac.il (M. Zehavi).

http://dx.doi.org/10.1016/j.tcs.2016.01.029

The language k-DISTINCT is a natural candidate for investigating relations between automata and parameterized problems. Indeed, we show that k-DISTINCT is implicitly related to several breakthrough techniques developed during the last two decades, to design algorithms for some fundamental parameterized problems. More precisely, building upon the color coding [2], divide-and-color [3] and narrow sieves [4] techniques, we obtain three different parameterized automata constructions for k-DISTINCT. Corresponding to the techniques on which they are based, these constructions allow developing deterministic, approximate counting and randomized parameterized algorithms.

We argue that developing parameterized algorithms using our constructions can be done in an easy, natural manner. In particular, for researchers knowledgeable with automata and graphs, our deterministic algorithm for k-PATH can be summarized by saying that, essentially, we just "take the product of the graph with the automaton for $L_k(V)$" (see, e.g., [5]), where V is the vertex set. To demonstrate our approach, we consider the following three problems.

Weighted k-path Given a directed graph $G=(V,E)$, a weight function $\psi: E \to \mathbb{R}$ and a parameter $k \in \mathbb{N}$, determine if G has a *simple k-path* (i.e., an acyclic/simple walk on k vertices), and if so, return such a path of minimum weight.

Weighted r-dimensional k-matching Given disjoint universes $U_1, \ldots, U_r$, the input consists of a family $\mathcal{S} \subseteq U_1 \times \ldots \times U_r$,[1] a weight function $\psi: \mathcal{S} \to \mathbb{R}$ and a parameter $k \in \mathbb{N}$. The goal is to determine if there exists a subfamily of k elements in $\mathcal{S}$ such that no element in $U_1 \cup \ldots \cup U_r$ is matched more than once, and if so, return such a subfamily of maximal weight.

Weighted module motif Let $G=(V,E)$ be an undirected graph, and let $k \in \mathbb{N}$. A *k-module* is a subset $U \subseteq V$ of k vertices that have the same neighbors outside of U. More formally, for every $u_1, u_2 \in U$ and $r \in V \setminus U$, $(u_1, r) \in E$ if and only if $(u_2, r) \in E$. Now, given a set C of colors, a function $Col: V \to 2^C$ specifying the admissible colors for each vertex, and a weight function $\psi: V \to \mathbb{R}$,[2] determine if there exists a k-module U in G and a coloring $col: U \to C$ such that

- for every $v \in U$, $col(v) \in Col(v)$, and
- for every $c \in C$, there is at most one vertex in U whose color is c (according to col),

and if so, return a pair that maximizes the weight $\psi(U)$ of the module U, defined as $\psi(U) \triangleq \sum_{v \in U} \psi(v)$.

We highlight two variants of the above problems, considered in this paper.

1. The *unweighted* version, where all elements are assumed to have weight zero.
2. The *approximate counting* unweighted version, where, given an *accuracy parameter* $\delta > 1$, the goal is to return a value in the interval $[S/\delta,\ \delta \cdot S]$, where S is the number of solutions (e.g., the number of simple k-paths).

1.1. Prior work

The k-PATH and r-DIMENSIONAL k-MATCHING problems are two of the most well-studied problems in the field of parameterized complexity. Indeed, the k-PATH problem has enjoyed a race towards obtaining the fastest parameterized algorithm for it [7,2,4,8,3,9–15]. Currently, the best known parameterized algorithm for WEIGHTED k-PATH runs in time $O^*(2.619^k)$ [14,10]. For (unweighted) k-PATH, Williams [15] gave a randomized algorithm running in time $O^*(2^k)$. Restricted to undirected graphs, k-PATH can be solved in randomized time $O^*(1.66^k)$ [4]. We also note that approximately counting the number of simple k-paths in a graph can be performed in time $O^*((2e)^{k+o(k)})$ [16].

The classic decision version of the r-DIMENSIONAL k-MATCHING problem is listed as one of the six fundamental NP-complete problems in Garey and Johnson [17]. A considerable number of papers presented parameterized algorithms for this problem [4,18,19,3,20–25,12,26–30]. Currently, the best known parameterized algorithm for WEIGHTED r-DIMENSIONAL k-MATCHING runs in time $O^*(2.851^{(r-1)k})$ [24]. For r-DIMENSIONAL k-MATCHING, Björklund et al. [4] gave a randomized algorithm running in time $O^*(2^{(r-2)k})$. We also note that approximately counting the number of 3-dimensional k-matchings can be performed in randomized time $O^*(5.48^{3k})$ [31].

The MODULE MOTIF problem was recently introduced by Rizzi et al. [32], primarily motivated from the analysis of complex biological networks. This problem can be solved in deterministic and randomized times $O^*(4.32^k)$ and $O^*(2^k)$, respectively [6].

1.2. Our contribution

Building upon previous techniques, we obtain useful automata constructions for k-DISTINCT. To this end, we isolate ingredients common to algorithms based on each technique from those that are, in a sense, problem-specific adjustments; then,

[1] That is, $\mathcal{S} \subseteq \{\{u_1, u_2, \ldots, u_r\} : u_i \in U_i \text{ for all } i \in \{1, 2, \ldots, r\}\}$.

[2] For the sake of clarity, we consider vertex weights, rather than color similarity scores (see [6]). Our algorithms can be easily modified to handle color similarity scores.