Journal of Computer and System Sciences ••• (••••) •••–•••

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

Reconfiguration on sparse graphs ☆

Daniel Lokshtanov [a], Amer E. Mouawad [a,*], Fahad Panolan [a], M.S. Ramanujan [b], Saket Saurabh [a,c]

[a] University of Bergen, Norway
[b] TU Wien, Vienna, Austria
[c] Institute of Mathematical Sciences, Chennai, India

ARTICLE INFO

Article history:
Received 24 August 2017
Received in revised form 22 February 2018
Accepted 26 February 2018
Available online xxxx

Keywords:
Reconfiguration
Dominating set
Independent set
Sparse graphs
Parameterized complexity

ABSTRACT

A vertex-subset graph problem $\mathcal{Q}$ defines which subsets of the vertices of an input graph are feasible solutions. A reconfiguration variant of a vertex-subset problem asks, given two feasible solutions of size k, whether it is possible to transform one into the other by a sequence of vertex additions/deletions such that each intermediate set remains a feasible solution of size bounded by k. We study reconfiguration variants of two classical vertex-subset problems, namely Independent Set and Dominating Set. We denote the former by ISR and the latter by DSR. Both ISR and DSR are PSPACE-complete on graphs of bounded bandwidth and W[1]-hard parameterized by k on general graphs. We show that ISR is fixed-parameter tractable parameterized by k when the input graph is of bounded degeneracy or nowhere dense. For DSR, we show the problem fixed-parameter tractable parameterized by k when the input graph does not contain large bicliques.

1. Introduction

Given an n-vertex graph G and two vertices s and t in G, determining whether there exists a path and computing the length of the shortest path between s and t are two of the most fundamental graph problems. In the classical battle of P versus NP or "easy" versus "hard", both of these problems are on the easy side. That is, they can be solved in $poly(n)$ time, where $poly$ is any polynomial function. But what if our input consisted of a 2^n-vertex graph? Of course, we can no longer assume G to be part of the input, as reading the input alone requires more than $poly(n)$ time. Instead, we are given an oracle encoded using $poly(n)$ bits and that can, in constant or $poly(n)$ time, answer queries of the form "is u a vertex in G" or "is there an edge between u and v?". Given such an oracle and two vertices of the 2^n-vertex graph, can we still determine if there is a path or compute the length of the shortest path between s and t in $poly(n)$ time?

This seemingly artificial question is in fact quite natural and appears in many practical and theoretical problems. In particular, these are exactly the types of questions asked under the reconfiguration framework, the main subject of this work. Under the reconfiguration framework, instead of finding a feasible solution to some instance $\mathcal{I}$ of a search problem $\mathcal{Q}$, we are interested in structural and algorithmic questions related to the solution space of $\mathcal{Q}$. Naturally, given some

☆ An extended abstract of this paper has appeared in the proceedings of the 14th International Symposium on Algorithms and Data Structures (WADS 2015). The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement n. 267959.

* Corresponding author.
E-mail addresses: daniello@ii.uib.no (D. Lokshtanov), a.mouawad@uib.no (A.E. Mouawad), fahad.panolan@uib.no (F. Panolan), ramanujan@ac.tuwien.ac.at (M.S. Ramanujan), saket@imsc.res.in (S. Saurabh).

https://doi.org/10.1016/j.jcss.2018.02.004

adjacency relation $\mathcal{A}$ defined over feasible solutions of $\mathcal{Q}$, the solution space can be represented using a graph $R_{\mathcal{Q}}(\mathcal{I})$, called the *reconfiguration graph*. $R_{\mathcal{Q}}(\mathcal{I})$ contains one node for each feasible solution of $\mathcal{Q}$ on instance $\mathcal{I}$ and two nodes share an edge whenever their corresponding solutions are adjacent under $\mathcal{A}$. An edge in $R_{\mathcal{Q}}(\mathcal{I})$ corresponds to a *reconfiguration step*, a walk in $R_{\mathcal{Q}}(\mathcal{I})$ is a sequence of such steps, a *reconfiguration sequence*.

Studying problems related to reconfiguration graphs has received considerable attention in the literature [1–6], the most popular problem being to determine whether there exists a reconfiguration sequence between two given feasible solutions/configurations. In many cases, this problem was shown PSPACE-hard in general, although some polynomial-time solvable restricted cases have been identified. For PSPACE-hard cases, it is not surprising that shortest paths between solutions can have exponential length. More surprising is that for most known polynomial-time solvable cases the diameter of the reconfiguration graph has been shown to be polynomial. Some of the problems that have been studied under the reconfiguration framework include Independent Set [7], Shortest Path [8], Coloring [9], Boolean Satisfiability [2], and Flip Distance [1,10]. We refer the reader to the survey by van den Heuvel [11] for a detailed overview of reconfiguration problems and their applications. A systematic study of the parameterized complexity [12] of reconfiguration problems was initiated by Mouawad et al. [6]; various problems were identified where the problem was not only NP-hard (or PSPACE-hard), but also W-hard under various parameterizations. The reader is referred to [12] for more on parameterized complexity and kernelization.

Overview of our results. In this work, we focus on reconfiguration variants of the Independent Set (IS) and Dominating Set (DS) problems. Given two independent sets I_s and I_t of a graph G such that $|I_s| = |I_t| = k$, the Independent Set Reconfiguration (ISR) problem asks whether there exists a sequence of independents sets $\sigma = \langle I_0, I_1, \ldots, I_\ell \rangle$, for some ℓ, such that:

(1) $I_0 = I_s$ and $I_\ell = I_t$,
(2) I_i is an independent set of G for all $0 \le i \le \ell$,
(3) $|\{I_i \setminus I_{i+1}\} \cup \{I_{i+1} \setminus I_i\}| = 1$ for all $0 \le i < \ell$, and
(4) $k - 1 \le |I_i| \le k$ for all $0 \le i \le \ell$.

Alternatively, given a graph G and integer k, the $R_{\text{IS}}(G, k-1, k)$ reconfiguration graph has a node for each independent set of G of size k or $k-1$ and two nodes are adjacent in $R_{\text{IS}}(G, k-1, k)$ whenever the corresponding independent sets can be obtained from one another by either the addition or the deletion of a single vertex. The reconfiguration graph $R_{\text{DS}}(G, k, k+1)$ is defined similarly for dominating sets. Hence, ISR and DSR can be formally stated as follows:

Independent Set Reconfiguration (ISR)
Input: Graph G, integer $k > 0$, and two independent sets I_s and I_t of size k
Question: Is there a path from I_s to I_t in $R_{\text{IS}}(G, k-1, k)$?

Dominating Set Reconfiguration (DSR)
Input: Graph G, integer $k > 0$, and two dominating sets D_s and D_t of size k
Question: Is there a path from D_s to D_t in $R_{\text{DS}}(G, k, k+1)$?

Note that since we only allow independent sets of size k and $k-1$ the ISR problem is equivalent to reconfiguration under the token jumping model considered by Ito et al. [13,14]. ISR is known to be PSPACE-complete on graphs of bounded bandwidth [15] (hence pathwidth and treewidth) and W[1]-hard when parameterized by k on general graphs [14]. On the positive side, the problem was shown fixed-parameter tractable, with parameter k, for graphs of bounded degree, planar graphs, and graphs excluding $K_{3,d}$ as a (not necessarily induced) subgraph, for any constant d [13,14]. We push this boundary further by showing that the problem remains fixed-parameter tractable for graphs of bounded degeneracy and nowhere dense graphs. As a corollary, we answer positively the question concerning the parameterized complexity of the problem parameterized by k on graphs of bounded treewidth.

For DSR, we show that the problem is fixed-parameter tractable, with parameter k, for graphs excluding $K_{d,d}$ as a (not necessarily induced) subgraph, for any constant d. Note that this class of graphs includes both nowhere dense and bounded degeneracy graphs and is the "largest" class on which the Dominating Set problem is known to be in FPT [16,17].

Our main open question, which was recently answered positively by Bousquet et al. [18], is whether ISR remains fixed-parameter tractable on graphs excluding $K_{d,d}$ as a subgraph. Also closely related is the work of Siebertz [19] who showed that for the distance-r variants of Independent Set and Dominating Set the reconfiguration problems become W[1]-hard on somewhere dense graphs. Specifically, if a class of graphs $\mathcal{C}$ is somewhere dense and closed under taking subgraphs, then for some value of $r \ge 1$ the reconfiguration problems are W[1]-hard. It remains to be seen whether we can adapt our results for ISR to find shortest reconfiguration sequences. Our algorithm for DSR does in fact guarantee shortest reconfiguration sequences but, as we shall see, the same does not hold for either of the two ISR algorithms.