When is the evaluation of conjunctive queries tractable ?

Martin Grohe*
University of Illinois at Chicago

Thomas Schwentick†
Friedrich-Schiller-Universität Jena

Luc Segoufin‡
INRIA-Rocquencourt

Abstract

The evaluation of conjunctive queries is hard both with respect to its *combined complexity* (NP-complete) and its *parameterized complexity* (W[1]-complete). It becomes tractable (PTIME for combined complexity, FPT for parameterized complexity), when the underlying graphs of the conjunctive queries have bounded *tree-width* [2]. We show that, in some sense, this is optimal both with respect to combined and parameterized complexity: For every class C of graphs, the evaluation of all conjunctive queries whose underlying graph is in C is tractable if, and only if, C has bounded tree-width.
A technical result of independent interest is that the *colored grid homomorphism* problem is NP-complete and, if parameterized by the grid size, W[1]-complete.

1. Introduction

Conjunctive queries are relational database queries expressed by formulas of first-order logic that are of the form

$$\exists y_1 \dots \exists y_l (\alpha_1 \wedge \dots \wedge \alpha_n),$$

where the α_i are atomic formulas. Conjunctive queries are fundamental for relational database systems, and they are the queries appearing most frequently in practice. The *conjunctive query evaluation* problem is the following: Given a finite relational structure (or database instance) $\mathcal{A}$ and a conjunctive query $\varphi(x_1, \dots, x_k)$, compute the set $\varphi(\mathcal{A}) := \{(a_1, \dots, a_k) \mid \mathcal{A} \models \varphi(a_1, \dots, a_k)\}$. In the complexity theoretic analysis of this paper, we will focus on the associated decision problem of whether $\varphi(\mathcal{A})$ is empty or not.

We will measure the complexity of query evaluation in two different ways. The most straightforward measure, where the complexity is measured simply in terms of the input size, i.e. size of the structure $\mathcal{A}$ plus length of the formula φ, is called the *combined complexity* of the query evaluation problem [14]. Chandra and Merlin [1] proved in 1977 that the combined complexity of conjunctive query evaluation is NP-complete. However, the main factor responsible for the high complexity of the problem is the length of the input formula and not the size of the database. This does not fit the practical situation very well, because usually one evaluates a short query against a large database. It is easy to see that conjunctive (and, more general, first-oder) queries can always be evaluated in time $||\mathcal{A}||^{O(||\varphi||)}$, where $||\mathcal{A}||$ denotes the size of the input structure $\mathcal{A}$ and $||\varphi||$ the length of the formula φ. Thus conjunctive query evaluation is possible in time polynomial in the structure size. However, even for short queries φ, say, of length $||\varphi|| = 10$, $||\mathcal{A}||^{||\varphi||}$ is not a feasible complexity. Yannakakis [16] was the first to suggest that *parameterized complexity* might be an appropriate way to measure the complexity of database query evaluation. We say that the query evaluation problem is *fixed-parameter tractable* (FPT) if there is a computable function $f : \mathbb{N} \to \mathbb{N}$ and a constant $c > 0$ such that the problem can be solved in time $f(||\varphi||) \cdot n^c$. The idea is that we can disregard the dependence on the formula length as long as it does not appear in the exponent of the structure size. Of course, the notion of fixed-parameter tractability is only another mathematical abstraction of "practical tractability", but in the situation of query evaluation it seems to be the most appropriate abstraction. Unfortunately, it turns out that the parameterized complexity of conjunctive query evaluation is W[1]-complete and thus most likely not fixed-parameter tractable; this result, observed by Papadimitriou and Yannakakis [11], can be seen as the parameterized analogue of the NP-completeness of the combined complexity.

The results mentioned in the previous paragraph show that conjunctive query evaluation is hard both with respect to its combined complexity and its parameterized complexity. Therefore, it is natural to look for classes of conjunctive queries for which the evaluation is tractable. The archetypal result of this form, due to Yannakakis [15], states that the evaluation of *acyclic* conjunctive queries is possible in polynomial time (and therefore both tractable in the classical sense of combined complexity and fixed-parameter tractable). This result has been improved in several ways.

*Department of Mathematics, Statistics & Computer Science, University of Illinois at Chicago, 322 SEO, 851 S. Morgan, Chicago, IL 60607-7045, U.S.A. Email: grohe@math.uic.edu.
†Institut für Informatik, Friedrich-Schiller-Universität Jena, D-07740 Jena, Germany. Email: tick@informatik.uni-jena.de
‡INRIA-Rocquencourt, B.P. 105, Le Chesnay Cedex 78153, France. Email: Luc.Segoufin@inria.fr.

One of the approaches for finding tractable queries is the following: With every conjunctive query φ we associate its *underlying graph* $\mathcal{G}_\varphi$ whose vertices are the variables of φ, with an edge between two variables x and y if there is an atom of φ that contains both x and y. Then the hope is that the evaluation of a query is easy if its underlying graph is "simple". *Tree-width*, measuring the similarity of a graph with a tree, turns out to be a good notion of simplicity. Chekuri and Rajaraman [2] have proved that evaluation of conjunctive queries of *bounded tree-width* is possible in polynomial time. (The result is presented here using a different formalization due to Kolaitis and Vardi [10]). Gottlob, Leone, and Scarcello [7] showed that this problem is actually LOGCFL-complete, for each fixed tree-width k. Other, similar approaches are based on the *underlying hypergraph* of a query. Appropriate notions of simplicity of queries obtained this way are bounded query-width [2] and bounded hypertree-width [8]. Although these notions are equivalent to bounded tree-width when the schema of the database is considered as fixed, the hypergraph based approach leads to more general classes of queries when the schema is not fixed (see [8] for a comparison of the different approaches).

In this paper, we focus on the graph based approach. We show that Chekuri and Rajaraman's result is optimal both with respect to combined complexity and parameterized complexity. Under the complexity theoretic assumption that FPT $\neq$ W[1], which can be seen as the analogue of PTIME $\neq$ NP in the parameterized world, we show that for every class C of graphs the following three statements are equivalent:

(1) The evaluation of all conjunctive queries whose underlying graph is in C is fixed-parameter tractable.

(2) The evaluation of all conjunctive queries whose underlying graph is in C is in polynomial time.

(3) The class C has bounded tree-width.

To prove this, we show that if C does not have bounded tree-width, the evaluation problem for conjunctive queries whose underlying graph is in C is W[1]-complete. This also yields a nice dichotomy: Either the evaluation problem for queries with underlying graph in C is W[1]-complete, or it is fixed-parameter tractable.

Another noteworthy aspect of this result is that the equivalence between (2) and (3) that is only concerned with classical complexity theory is proved under an assumption from parameterized complexity theory. Under the weaker assumption that PTIME $\neq$ NP, we can only prove the equivalence of statements (2) and (3) for all classes of graphs that are closed under taking minors.

In the last section of the paper, we extend our results to the existential fragment of first-order logic by showing that tractability of query evaluation is linked to an expressive power limited by the number of variables.

The main ingredient of our proofs is of independent technical interest: We show that the *colored grid homomorphism* problem, asking whether a given 2-colored grid can be mapped homomorphically to a given colored graph is NP-complete and, if parameterized by the grid size, W[1]-complete. We also prove a similar result for directed grids. These results are linked to tree-width by the deep Excluded Grid Theorem due to Robertson and Seymour [12].

2. Preliminaries

Relational Structures. A *vocabulary* is a finite set of relation symbols. The *arity* of a vocabulary is the maximum of the arities of all relations symbols it contains. In the following, τ always denotes a vocabulary. E always denotes a binary relation symbol and $C_1, C_2, \dots$ denote unary relation symbols. A *τ-structure* $\mathcal{A}$ consists of a non-empty set A, called the *universe* of $\mathcal{A}$, and a relation $R^{\mathcal{A}} \subseteq A^r$ for each r-ary relation symbol $R \in \tau$. *In this paper we only consider finite structures.* If C is a class of structures, C$[\tau]$ denotes the subclass of all τ-structures in C. If $\mathcal{A}$ is a τ-structure and $B \subseteq A$, then $\langle B \rangle^{\mathcal{A}}$ denotes the substructure induced by $\mathcal{A}$ on B, i.e. the τ-structure $\mathcal{B}$ with universe B and $R^{\mathcal{B}} = R^{\mathcal{A}} \cap B^r$ for every r-ary $R \in \tau$. A graph is an $\{E\}$-structure $\mathcal{G} = (G, E^{\mathcal{G}})$, where the binary relation $E^{\mathcal{G}}$ is symmetric and anti-reflexive (i.e. graphs are undirected and loop-free). A *directed graph* is an $\{E\}$-structure $\mathcal{D} = (D, E^{\mathcal{D}})$, where $E^{\mathcal{D}}$ is anti-reflexive. A *c-colored graph*, for a $c \geq 1$, is an $\{E, C_1, \dots, C_c\}$-structure $\mathcal{G}$, where $(G, E^{\mathcal{G}})$ is a graph and $C_1^{\mathcal{G}}, \dots, C_c^{\mathcal{G}}$ are disjoint sets (possibly empty) whose union is G. If $\mathcal{G}$ is a c-colored graph and $a \in G$ we define the mapping $\mathrm{col}^{\mathcal{G}} : G \to \{1, \dots, c\}$ by $\mathrm{col}^{\mathcal{G}}(a) = i$ if $a \in C_i^{\mathcal{G}}$. The distance $d^{\mathcal{G}}(a, b)$ between two points a, b in a graph or colored graph is the length of the shortest path from a to b.

For $m, n \geq 1$, the *(m,n)-grid* is the graph $\mathcal{G}_{m,n}$ with universe $G_{m,n} := \{1, \dots, m\} \times \{1, \dots, n\}$ and edge relation

$$E^{\mathcal{G}_{m,n}} := \Big\{ \big((i,j),(i',j')\big) \in G_{m,n}^2 \;\Big|\; (i = i' \text{ and } |j - j'| = 1) \text{ or } (|i - i'| = 1 \text{ and } j = j') \Big\}.$$

If $m = n$, then we call $\mathcal{G}_{m,n}$ a *square grid*.

A *homomorphism* from a τ-structure $\mathcal{A}$ to a τ-structure $\mathcal{B}$ is a mapping $h : A \to B$ such that for every $R \in \tau$ and $(a_1, \dots, a_r) \in R^{\mathcal{A}}$ we have $(h(a_1), \dots, h(a_r)) \in R^{\mathcal{B}}$. HOM$(\mathcal{A}, \mathcal{B})$ is the set of all homomorphisms from $\mathcal{A}$ to $\mathcal{B}$.

For every class C of structures, we consider the following problem:

C-HOMOMORPHISM	
Input:	Structures $\mathcal{A} \in$ C, $\mathcal{B}$.
Problem:	Is there a homomorphism from $\mathcal{A}$ to $\mathcal{B}$.

It is well known that GRAPH-HOMOMORPHISM is NP-complete (where GRAPH denotes the class of all graphs).

Parameterized complexity. A *parameterized problem* is a set $P \subseteq \Sigma^* \times \Pi^*$, where Σ and Π are finite alphabets. We usually represent a parameterized problem P in the following form:

Input:	$I \in \Sigma^*$.
Parameter:	$\pi \in \Pi^*$
Problem:	Decide if $(I, \pi) \in P$.

In this paper, we always have $\Pi = \{0, 1\}$ and consider the parameters $\pi \in \Pi^*$ as natural numbers (in binary).

Definition 1. A parameterized problem $P \subseteq \Sigma^* \times \Pi^*$ is *fixed-parameter tractable* if there is a computable function $f : \Pi^* \to \mathbb{N}$, a constant $c \in \mathbb{N}$ and an algorithm that, given $(I, \pi) \in \Sigma^* \times \Pi^*$, decides if $(I, \pi) \in P$ in time $f(\pi)|I|^c$. FPT denote the class of all fixed-parameter tractable parameterized problems.

Definition 2. Let $P \subseteq \Sigma^* \times \Pi^*$ and $P' \subseteq (\Sigma')^* \times (\Pi')^*$ be parameterized problems.

(1) A *parameterized T-reduction* from P to P' is an algorithm with an oracle for P' that solves any instance (x, y) of P in time $f(|y|) \cdot |x|^c$ in such a way that for all questions "$(x', y') \in P'$?" to the oracle we have $|y'| \leq g(|y|)$ (for computable functions $f, g : \mathbb{N} \to \mathbb{N}$ and a constant $c \in \mathbb{N}$). P is *fixed-parameter T-reducible* to P' (we write $P \leq^{\mathrm{fp}}_{\mathrm{T}} P'$), if there is a parameterized T-reduction from P to P'.

(2) A *parameterized m-reduction* from P to P' is an algorithm that computes for every instance (x, y) of P an instance (x', y') of P' in time $f(|y|) \cdot |x|^c$ such that $|y'| \leq g(|y|)$ and $(x, y) \in P$ if, and only if, $(x', y') \in P'$ (for computable functions $f, g : \mathbb{N} \to \mathbb{N}$ and a constant $c \in \mathbb{N}$). P is *fixed-parameter m-reducible* to P' (we write $P \leq^{\mathrm{fp}}_{\mathrm{m}} P'$), if there is a parameterized m-reduction from P to P'.

Observe that $\leq^{\mathrm{fp}}_{\mathrm{T}}$ and $\leq^{\mathrm{fp}}_{\mathrm{m}}$ are transitive and that $P \leq^{\mathrm{fp}}_{\mathrm{m}} P'$ implies $P \leq^{\mathrm{fp}}_{\mathrm{T}} P'$. Furthermore, if $P \leq^{\mathrm{fp}}_{\mathrm{T}} P'$ and $P' \in$ FPT then $P \in$ FPT. We define *hardness* and *completeness* of parameterized problems for a parameterized complexity class (under parameterized m- or T-reductions) in the usual way. If we do not specify the reduction-type in a hardness-result, we always refer to m-reductions.
Downey and Fellows defined a hierarchy $\mathrm{W}[1] \subseteq \mathrm{W}[2] \subseteq \cdots$ of parameterized complexity classes, and they conjecture that this hierarchy is strict and that FPT is strictly contained in W[1]. The classes of the W-hierarchy are defined in terms of a parameterized version of the satisfiability problem for bounded-depth Boolean circuits. We refer the reader to [4] for the technical definitions. In this paper, we are only interested in the class W[1], which can be seen as an analogue of NP in parameterized complexity theory. Our results are based on the following Theorem 1.
Remember that the CLIQUE-problem is one of the basic NP-complete problems [9]:

CLIQUE	
Input:	Graph $\mathcal{G}$, integer $k \geq 1$.
Problem:	Decide if $\mathcal{G}$ has a k-clique.

Its parameterized version

PARAMETERIZED CLIQUE	
Input:	Graph $\mathcal{G}$, integer $k \geq 1$.
Parameter:	k
Problem:	Decide if $\mathcal{G}$ has a k-clique.

plays a similar role in parameterized complexity theory.

Theorem 1 (Downey and Fellows [3]). PARAMETERIZED CLIQUE *is* W[1]*-complete (under parameterized m-reductions).*

For any class C of structures, consider the following parameterized version of the homomorphism problem:

PARAMETERIZED C-HOMOMORPHISM	
Input:	Structures $\mathcal{A} \in$ C, $\mathcal{B}$.
Parameter:	$\|\|\mathcal{A}\|\|$
Problem:	Decide if there is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$.

PARAMETERIZED C-HOMOMORPHISM is in W[1] for every C. By a trivial reduction from PARAMETERIZED CLIQUE, it can be seen that PARAMETERIZED GRAPH-HOMOMORPHISM is W[1]-complete. The following lemma will be needed later:

Lemma 2. *Let* C *be a class of structures. Then the following problem is fixed-parameter T-reducible to* PARAMETERIZED C-HOMOMORPHISM.

PARAMETERIZED C-HOMOMORPHISM CONSTRUCTION	
Input:	Structures $\mathcal{A} \in$ C and $\mathcal{B}$.
Parameter:	$\|\|\mathcal{A}\|\|$
Problem:	Decide if there is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$ and, if this is the case, compute such a homomorphism.

Proof: Given $\mathcal{A} \in$ C and $\mathcal{B}$, we first check if there is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$. If this is not the case, we reject.
Otherwise, if $|B| \leq |A|$ we can find a homomorphism from $\mathcal{A}$ to $\mathcal{B}$ by exhaustive search, the time this requires can be bounded by a function only depending on $||\mathcal{A}||$. If $|B| > |A|$, there is a $b \in B$ such that there is a homomorphism from $\mathcal{A}$ to $\langle B \setminus \{b\} \rangle^{\mathcal{B}}$. By simply testing all b, we can find such a b and remove it from $\mathcal{B}$. We repeat this procedure until we arrive at a substructure $\mathcal{B}' \subseteq \mathcal{B}$ of size at most $|A|$ such that there still is a homomorphism from $\mathcal{A}$ to $\mathcal{B}'$. Now we can find such a homomorphism by exhaustive search. □

Corollary 3. *If* PARAMETERIZED C-HOMOMORPHISM *is in* FPT *then* PARAMETERIZED C-HOMOMORPHISM CONSTRUCTION *is also in* FPT.

Remark 1. In problems like PARAMETERIZED C-HOMOMORPHISM, the class C is not necessarily computable. When we say that PARAMETERIZED C-HOMOMORPHISM is in FPT for a non-computable C, we mean that there is a computable function $f : \mathbb{N} \to \mathbb{N}$, a constant $c \geq 1$, and an algorithm that, given structures $\mathcal{A}$ and $\mathcal{B}$, decides if there is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$ in time $f(||\mathcal{A}||) \cdot ||\mathcal{B}||^c$ if $\mathcal{A}$ happens to be in C. If $\mathcal{A} \notin$ C, the answer given by the algorithm may be wrong. (However, since f is computable we can always assume that the algorithm stops after $g(||\mathcal{A}||) \cdot ||\mathcal{B}||^c$ steps, for some computable function g.) Similarly, we can define what it means that PARAMETERIZED C-HOMOMORPHISM is W[1]-complete under parameterized m-reductions or parameterized T-reductions.

Tree-decompositions and minors. A *tree* is an acyclic graph. A *tree-decomposition* of a τ-structure $\mathcal{A}$ is a pair $(\mathcal{T}, (B_t)_{t \in T})$, where $\mathcal{T}$ is a tree and $(B_t)_{t \in T}$ a family of subsets of A (the *blocks* of the decomposition) such that

(1) For every $a \in A$, the set $\{t \in T \mid a \in B_t\}$ is non-empty and connected in $\mathcal{T}$ (that is, induces a subtree).

(2) For every $R \in \tau$ and all $\bar{a} \in R^{\mathcal{A}}$ there is a $t \in T$ such that $\bar{a} \in B_t$.

The *width* of a tree-decomposition $(\mathcal{T}, (B_t)_{t \in T})$ is $\max\{|B_t| \mid t \in T\} - 1$. The *tree-width* $\mathrm{tw}(\mathcal{A})$ of $\mathcal{A}$ is the minimum taken over the widths of all tree-decompositions of $\mathcal{A}$.
A class C of structures has *bounded tree-width* if there is a k such that $\mathrm{tw}(\mathcal{A}) \leq k$ for all $\mathcal{A} \in \mathrm{C}$.
A *minor* of a graph $\mathcal{G}$ is a graph $\mathcal{H}$ that is obtained from a subgraph of $\mathcal{G}$ by contracting edges. Equivalently, $\mathcal{H}$ is a minor of $\mathcal{G}$ if there is a mapping $\mu : H \to \mathrm{Pow}(G)$ such that for all $a \in H$ the subgraph $\langle \mu(a) \rangle^{\mathcal{G}}$ is connected, and for all $(a,b) \in E^{\mathcal{H}}$ there are $a' \in \mu(a), b' \in \mu(b)$ such that $(a',b') \in E^{\mathcal{G}}$. We call μ a *minor map* from $\mathcal{H}$ to $\mathcal{G}$. We write $\mathcal{H} \preceq \mathcal{G}$ to denote that $\mathcal{H}$ is a minor of $\mathcal{G}$ and $\mu : \mathcal{H} \preceq \mathcal{G}$ to denote that μ is a minor map from $\mathcal{H}$ to $\mathcal{G}$.
The connection between grids and tree-width is made by the following deep *Excluded Grid Theorem*:

Theorem 4 (Robertson and Seymour [12]). *Let* C *be a class of graphs. Then* C *has bounded tree-width if, and only if, there is a grid that is not a minor of any graph in* C.

3. Grid homomorphism

For $c \geq 1$, let c-GRID be the class of all c-colored square grids. We shall prove that c-GRID HOMOMORPHISM is NP-complete and PARAMETERIZED c-GRID HOMOMORPHISM is W[1]-complete for every $c \geq 2$.
Before we show this, we observe that 1-GRID HOMOMORPHISM is trivially in PTIME, because as a bi-partite graph every grid has a homomorphism to a single edge. We also observe that, as an instance of the general HOMOMORPHISM problem for relational structures, c-GRID HOMOMORPHISM is contained in NP, and its parameterized version is contained in W[1]. Thus we just have to prove hardness. Of course it suffices to consider the case $c = 2$ (which, by the way, is considerably harder than $c \geq 3$).
Let $n \geq 6$ be even. We let $\mathcal{S}_n$ be the 2-colored $(n \times n)$-grid with (cf. Figure 1(a))

$$\begin{aligned} C_1^{\mathcal{S}_n} :=& \big\{(1,1),(1,2),(1,n),(2,1),(4,1),(n,1)\big\} \\ &\cup \Big\{(i,j) \in S_n \;\Big|\; i+j \geq n+4, \big((i+j-n) \equiv 0 \,(\mathrm{mod}\ 4) \\ &\qquad \text{or } (i+j-n) \equiv 1 \,(\mathrm{mod}\ 4)\big)\Big\}, \\ C_2^{\mathcal{S}_n} :=& S_n \setminus C_1^{\mathcal{S}_n}. \end{aligned}$$

The first major step in the proof of our completeness results is the following lemma:

Lemma 5. $\mathrm{HOM}(\mathcal{S}_n, \mathcal{S}_n) = \{\mathrm{id}\}$ *(i.e. the only homomorphism from $\mathcal{S}_n$ to $\mathcal{S}_n$ is the identity).*

The proof requires some preparation. For every $m \geq 1$ we let $\mathcal{P}_m$ be the 2-colored path with universe $P_m := \{1, \dots, m\}$, edge relation $E^{\mathcal{P}_m} := \big\{(i,j) \;\big|\; |i-j| = 1\big\}$, and

$$\begin{aligned} C_1^{\mathcal{P}_m} :=& \big\{i \in P_m \;\big|\; i \equiv 1 \,(\mathrm{mod}\ 4) \text{ or } i \equiv 2 \,(\mathrm{mod}\ 4)\big\}, \\ C_2^{\mathcal{P}_m} :=& P_m \setminus C_1^{\mathcal{P}_m}. \end{aligned}$$

$$\begin{array}{ccccccccccccccc}
\mathbf{1} & - & \mathbf{1} & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 & - & \mathbf{1} \\
| & & | & & | & & | & & | & & | & & | & & | \\
\mathbf{1} & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 \\
| & & | & & | & & | & & | & & | & & | & & | \\
2 & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 \\
| & & | & & | & & | & & | & & | & & | & & | \\
\mathbf{1} & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 & - & \mathbf{1} \\
| & & | & & | & & | & & | & & | & & | & & | \\
2 & - & 2 & - & 2 & - & 2 & - & 2 & - & 2 & - & \mathbf{1} & - & \mathbf{1} \\
| & & | & & | & & | & & | & & | & & | & & | \\
2 & - & 2 & - & 2 & - & 2 & - & 2 & - & \mathbf{1} & - & \mathbf{1} & - & 2 \\
| & & | & & | & & | & & | & & | & & | & & | \\
2 & - & 2 & - & 2 & - & 2 & - & \mathbf{1} & - & \mathbf{1} & - & 2 & - & 2 \\
| & & | & & | & & | & & | & & | & & | & & | \\
\mathbf{1} & - & 2 & - & 2 & - & \mathbf{1} & - & \mathbf{1} & - & 2 & - & 2 & - & \mathbf{1}
\end{array}$$

(a) The colored grid $\mathcal{S}_8$

$$\begin{array}{ccccccccccccccc}
1 & - & 2 & - & 3 & - & 4 & - & 3 & - & 4 & - & 3 & - & 2 \\
| & & | & & | & & | & & | & & | & & | & & | \\
2 & - & 3 & - & 4 & - & 3 & - & 4 & - & 3 & - & 4 & - & 3 \\
| & & | & & | & & | & & | & & | & & | & & | \\
3 & - & 4 & - & 3 & - & 4 & - & 3 & - & 4 & - & 3 & - & 4 \\
| & & | & & | & & | & & | & & | & & | & & | \\
2 & - & 3 & - & 4 & - & 3 & - & 4 & - & 3 & - & 4 & - & 5 \\
| & & | & & | & & | & & | & & | & & | & & | \\
3 & - & 4 & - & 3 & - & 4 & - & 3 & - & 4 & - & 5 & - & 6 \\
| & & | & & | & & | & & | & & | & & | & & | \\
4 & - & 3 & - & 4 & - & 3 & - & 4 & - & 5 & - & 6 & - & 7 \\
| & & | & & | & & | & & | & & | & & | & & | \\
3 & - & 4 & - & 3 & - & 4 & - & 5 & - & 6 & - & 7 & - & 8 \\
| & & | & & | & & | & & | & & | & & | & & | \\
2 & - & 3 & - & 4 & - & 5 & - & 6 & - & 7 & - & 8 & - & 9
\end{array}$$

(b) The colored grid $\mathcal{S}'_8$

Figure 1.

Let $X := \{(1,1),(2,1),(4,1),(n,1),(1,n)\}$. We define a mapping $h_n : S_n \to P_{n+1}$ by

$$h_n((i,j)) := \begin{cases} 1 & \text{if } (i,j) = (1,1), \\ 2 & \text{if } (i,j) \in X \setminus \{(1,1)\}, \\ 3 & \text{if } (i,j) \notin X \\ & \quad \text{and } i+j < n+4 \\ & \quad \text{and } i+j \equiv 0 \,(\mathrm{mod}\ 2), \\ 4 & \text{if } (i,j) \notin X \\ & \quad \text{and } i+j < n+4 \\ & \quad \text{and } i+j \equiv 1 \,(\mathrm{mod}\ 2), \\ i+j-n+1 & \text{if } i+j \geq n+4 \end{cases}$$

(cf. Figure 1(b)). It is easy to verify that

$$h_n \in \mathrm{HOM}(\mathcal{S}_n, \mathcal{P}_{n+1}).$$

Lemma 6. $\mathrm{HOM}(\mathcal{S}_n, \mathcal{P}_{n+1}) = \{h_n\}$.

Proof: Let $h : \mathcal{S}_n \to \mathcal{P}_{n+1}$ be an arbitrary homomorphism. We shall prove that $h = h_n$. Let $i_1 := h((1,1))$. Then $i_1 \in C_1^{\mathcal{P}_{n+1}}$, and thus $i_1 \equiv 1 \,(\mathrm{mod}\ 4)$ or $i_1 \equiv 2 \,(\mathrm{mod}\ 4)$.
Case 1: $i_1 \equiv 1 \,(\mathrm{mod}\ 4)$.

Then $h((1,2)) = h((2,1)) = i_1 + 1$. Since $(4,1)$, $(n,1)$, and $(1,n)$ are reachable in $\mathcal{S}_n$ from $(2,1)$ by paths of even length whose inner vertices are all in $\mathcal{C}_2^{\mathcal{S}_n}$, we must have $h((4,1)) = h((n,1)) = h((1,n)) = i_1 + 1$. Similarly, all (i,j) with $(i,j) \notin X$, $i+j < n+4$, and $i+j \equiv 0 \pmod 2$ are reachable in $\mathcal{S}_n$ from $(2,1)$ by paths of odd length whose inner vertices are all in $\mathcal{C}_2^{\mathcal{S}_n}$. Hence we have $h((i,j)) = i_1+2$. All (i,j) with $(i,j) \notin X$, $i+j < n+4$, and $i+j \equiv 1 \pmod 2$ are reachable in $\mathcal{S}_n$ from $(2,1)$ by paths of even length whose inner vertices are all in $\mathcal{C}_2^{\mathcal{S}_n}$. Hence we have $h((i,j)) = i_1+3$. All (i,j) with $(i,j) \notin X$ and $i+j = n+4$ are reachable in $\mathcal{S}_n$ from $(2,1)$ by paths of odd length whose inner vertices are all in $\mathcal{C}_2^{\mathcal{S}_n}$. Hence we have $h((i,j)) = i_1 + 4$. Now a simple induction shows that for all (i,j) with $i+j \geq n+4$ we have $h((i,j)) = i_1 + i + j - n$.
Altogether, this shows that $i_1 = 1$ and $h = h_n$.

Case 2: $i_1 \equiv 2 \pmod 4$.
We argue similarly as in Case 1 with the direction of the path reversed and show that $i_1 = n+1$. Then $h((2,1)) = n$, which implies $1 = \mathrm{col}^{\mathcal{S}_n}((2,1)) = \mathrm{col}^{\mathcal{P}_n}(n)$. But this immediately yields a contradiction since $n \equiv 0 \pmod 4$ and thus $\mathrm{col}^{\mathcal{P}_n}(n) = 2$. □

We now define an $(n+1)$-colored (n,n)-grid $\mathcal{S}_n'$ by letting $\mathrm{col}^{\mathcal{S}_n'}(a) = h_n(a)$ for all $a \in S_n' = S_n$.

Lemma 7. $\mathrm{HOM}(\mathcal{S}_n, \mathcal{S}_n) = \mathrm{HOM}(\mathcal{S}_n', \mathcal{S}_n')$.

Proof: $\mathrm{HOM}(\mathcal{S}_n', \mathcal{S}_n') \subseteq \mathrm{HOM}(\mathcal{S}_n, \mathcal{S}_n)$ is trivial. To see the converse, let $g : \mathcal{S}_n \to \mathcal{S}_n$ be a homomorphism. Then for all $a \in S_n$ we have $\mathrm{col}^{\mathcal{S}_n'}(g(a)) = h_n(g(a)) = h_n(a) = \mathrm{col}^{\mathcal{S}_n'}(a)$, since $h_n \circ g = h_n$ by Lemma 6. □

Proof (of Lemma 5): By the previous lemma, it suffices to show that $\mathrm{HOM}(\mathcal{S}_n', \mathcal{S}_n') = \{\mathrm{id}\}$. So let $h \in \mathrm{HOM}(\mathcal{S}_n', \mathcal{S}_n')$. We will repeatedly use the following observation, holding for all $a, b \in S_n'$:

$$d^{\mathcal{S}_n'}(a,b) \geq d^{\mathcal{S}_n'}(h(a), h(b)) \tag{1}$$

We have $h((1,1)) = (1,1)$ and $h((n,n)) = (n,n)$, because $(1,1)$ and (n,n) are the only vertices of their respective colors. Now we consider the vertices $(n,1)$ and $(1,n)$. Since they are the only vertices $a \in C_2^{\mathcal{S}_n'}$ with $d^{\mathcal{S}_n'}((1,1),a) \leq n-1$ and $d^{\mathcal{S}_n'}((n,n),a) \leq n-1$, we have $h((n,1)), h((1,n)) \in \{(n,1),(1,n)\}$. Remember that $(4,1) \in C_2^{\mathcal{S}_n'}$, and note that $d^{\mathcal{S}_n'}((1,1),(4,1)) = 3$ and $d^{\mathcal{S}_n'}((n,1),(4,1)) = n-4$. Since there is no $a \in C_2^{\mathcal{S}_n'}$ with $d^{\mathcal{S}_n'}((1,1),a) \leq 3$, $d^{\mathcal{S}_n'}((1,n),a) \leq n-4$, we must have $h((n,1)) = (n,1)$. Similarly, since there is no $a \in C_4^{\mathcal{S}_n'}$ with $d^{\mathcal{S}_n'}((1,1),a) \leq 3$, $d^{\mathcal{S}_n'}((n,1),a) \leq n-4$, we have $h((1,n)) = (1,n)$.
Thus h is the identity on the four corner points of $\mathcal{S}_n'$. Now a simple induction using (1) shows that $h = \mathrm{id}$. □

Let $k \geq 4$ such that $k \equiv 0 \pmod 4$ or $k \equiv 1 \pmod 4$. Then $K := \binom{k}{2}$ is even. Let $f : \{1, \dots, K\} \to \{\{i,j\} \mid 1 \leq i < j \leq k\}$ be the bijection that maps l to the lth element in the lexicographic order of $\{\{i,j\} \mid 1 \leq i < j \leq k\}$. Let $\mathcal{H}$ be an arbitrary graph. For $a \in H$ and $e = (b,c) \in E^{\mathcal{H}}$ we write $a \in e$ to denote that a is an endpoint of e, i.e. that $a \in \{b,c\}$. We define a 2-colored graph $\mathcal{H}_k$ as follows:

$$\begin{aligned}
H_k :=& \Big\{ (a,e,i,p) \;\Big|\; a \in H, e \in E^{\mathcal{H}}, 1 \leq i \leq k, 1 \leq p \leq K \\
&\quad \text{such that } \big(i \in f(p) \iff a \in e\big) \Big\} \\
&\cup \big\{ (i,p) \mid k+1 \leq i \leq K, 1 \leq p \leq K \big\}, \\
E^{\mathcal{H}_k} :=& \Big\{ \big((a,e,i,p),(a',e,i',p)\big) \;\Big|\; a, a' \in H, e \in E^{\mathcal{H}}, \\
&\quad 1 \leq i, i' \leq k, |i-i'| = 1, 1 \leq p \leq K \Big\} \\
&\cup \Big\{ \big((a,e,i,p),(a,e',i,p')\big) \;\Big|\; a \in H, e, e' \in E^{\mathcal{H}}, \\
&\quad 1 \leq i \leq k, 1 \leq p, p' < K, |p-p'| = 1 \Big\} \\
&\cup \Big\{ \big((a,e,k,p),(k+1,p)\big), \big((k+1,p),(a,e,k,p)\big) \;\Big| \\
&\quad a \in H, e \in E^{\mathcal{H}}, 1 \leq p \leq K \Big\} \\
&\cup \Big\{ \big((i,p),(i',p')\big) \;\Big|\; \big((i,p),(i',p')\big) \in E^{\mathcal{S}_K}, \\
&\quad k+1 \leq i \leq K, 1 \leq p \leq K \Big\}, \\
C_j^{\mathcal{H}_k} :=& \big\{ (a,e,i,p) \mid a \in H, e \in E^{\mathcal{H}}, 1 \leq i \leq k, 1 \leq p \leq K, \\
&\quad (i,p) \in C_j^{\mathcal{S}_K} \big\} \\
&\cup \big\{ (i,p) \mid k+1 \leq i \leq K, 1 \leq p \leq K, (i,p) \in C_j^{\mathcal{S}_K} \big\} \\
&\qquad\qquad (\text{for } j \in \{1,2\}).
\end{aligned}$$

The following lemma is obvious:

Lemma 8. *The mapping $\pi : \mathcal{H}_k \to \mathcal{S}_K$ defined by*

$$\pi((a,e,i,p)) = (i,p) \text{ and } \pi((i,p)) = (i,p)$$

is a homomorphism.

Lemma 9. *$\mathcal{H}$ contains a k-clique if, and only if, there is a homomorphism $h : \mathcal{S}_K \to \mathcal{H}_k$.*

Proof: For the forward direction, let $\{a_1, \dots, a_k\}$ be a clique of size k in $\mathcal{H}$, and for $1 \leq i < j \leq k$ let $e_{\{ij\}}$ denote the edge between a_i and a_j. Then we define $h : \mathcal{S}_K \to \mathcal{H}_k$ by

$$h((i,p)) = \begin{cases} (a_i, e_{f(p)}, i, p) & \text{if } 1 \leq i \leq k \\ (i,p) & \text{otherwise.} \end{cases}$$

It is easy to check that h is a homomorphism. Note that the main point is to verify that h is well-defined.

For the backward direction, suppose that $h : \mathcal{S}_K \to \mathcal{H}_k$ is a homomorphism. Then by Lemma 8, $\pi \circ h : \mathcal{S}_K \to \mathcal{S}_K$ is a homomorphism. Thus, recalling that $K \geq 6$ is even, by Lemma 5, $\pi \circ h$ is the identity. Hence for all $1 \leq i \leq k, 1 \leq p \leq K$ there are a_{ip}, e_{ip} such that $h(i,p) = (a_{ip}, e_{ip}, i, p)$. By the definition of $E^{\mathcal{H}_k}$, this implies $a_{ip} = a_{iq} =: a_i$ for $1 \leq i \leq k$, $1 \leq p, q \leq K$ and $e_{ip} = e_{jp} =: e_p$ for $1 \leq i, j \leq k$, $1 \leq p \leq K$.
Then by the definition of H_k, for $1 \leq i \leq k$ and $1 \leq p \leq K$ we have $a_i \in e_p$ if, and only if, $i \in f(p)$. Thus $e_{f^{-1}(\{i,j\})}$ is an edge between a_i and a_j, and therefore $\{a_1, \dots, a_k\}$ is a clique of $\mathcal{H}$. □

The following lemma is again obvious:

Lemma 10. *There is a polynomial time algorithm that, given a graph $\mathcal{H}$ and a k with $4 \leq k \leq |H|$ and $k \equiv 0, 1 \pmod 4$, computes $\mathcal{S}_K$ and $\mathcal{H}_k$.*

Thus the NP-completeness of CLIQUE and the W[1]-completeness of PARAMETERIZED CLIQUE immediately yield the following Theorem 11. There is a slight problem in that we only have a reduction for $k \geq 4$ with $k \equiv 0, 1 \pmod 4$. However, we can use the simple fact that for all $l < k$, a graph $\mathcal{H}$ has an l-clique if, and only if, the graph obtained from $\mathcal{H}$ by adding $(k-l)$ vertices and connecting them with each other and all old vertices has a k-clique.

Theorem 11. *Let $c \geq 2$. Then c-*GRID HOMOMORPHISM *is* NP*-complete. Furthermore,* PARAMETERIZED c-GRID HOMOMORPHISM *is* W[1]*-complete.*

Directed grids. Let DGRID denote the class of all directed graphs whose underlying graph is a square grid. We show how our proof can be modified to prove NP-completeness of DGRID-HOMOMORPHISM and W[1]-completeness of its parameterized version.

Lemma 12. *Let $\mathcal{D}$ be a directed acyclic graph with a directed Hamiltonian path. Then* $\mathrm{HOM}(\mathcal{D}, \mathcal{D}) = \{\mathrm{id}\}$.

Proof: Let $\mathcal{H}$ be a Hamiltonian path of $\mathcal{D}$. Define a linear order $\leq$ on D by letting $a \leq b$ if, and only if, a occurs before b on $\mathcal{H}$. Suppose that $h : \mathcal{D} \to \mathcal{D}$ is not the identity. Then there exist $a, b \in D$ such that $a \leq b$ and $h(b) < h(a)$. Then the interval of $\mathcal{H}$ that connects a with b must be mapped to a directed path from $h(a)$ to $h(b)$, which is impossible because $\mathcal{D}$ is acyclic. □

For every $n \geq 1$ we let $\mathcal{S}_n^d$ be the directed acyclic $(n \times n)$-gridthat is obtained by directing the edges of $\mathcal{G}_{n,n}$ according to the Hamiltonian path (cf. Figure 2).

$$(1,1), (1,2), \dots, (1,n), (2,n), \dots, (2,1), (3,1), \ \dots\ , (n,n)$$

Then it is an immediate consequence of Lemma 12 that

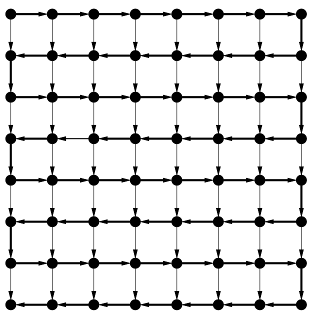

Figure 2. The directed grid $\mathcal{S}_8^d$

$\mathrm{HOM}(\mathcal{S}_n^d, \mathcal{S}_n^d) = \{\mathrm{id}\}$ for all $n \geq 1$.
We can now proceed completely analogously to the case of 2-colored grids. For every $k \geq 2$ and every undirected graph $\mathcal{H}$ we define a directed graph $\mathcal{H}_k^d$ and prove that $\mathcal{H}$ contains a k-clique if, and only if, there is a homomorphism from $\mathcal{S}_K^d$ to $\mathcal{H}_k^d$. Moreover, we do this in such a way that, given $\mathcal{H}$ and k, $\mathcal{S}_K^d$ and $\mathcal{H}_k^d$ can be computed in polynomial time. Thus we obtain:

Theorem 13. DGRID-HOMOMORPHISM *is* NP*-complete. Furthermore,* PARAMETERIZED DGRID-HOMOMORPHISM *is* W[1]*-complete.*

4. Conjunctive Query Evaluation

Atomic formulas, or *atoms*, are expressions of the form $x = y$ or $Rx_1 \dots x_r$, where R is an r-ary relation symbol and $x, y, x_1, \dots, x_r$ are *variables.* The formulas of *first-order logic* FO are build up in the usual way from the atomic formulas using the connectives $\wedge, \vee, \neg, \rightarrow$, and the quantifiers $\forall, \exists$.
The *vocabulary* of a first-order formula φ, denoted by $\tau(\varphi)$, is the set of all relation symbols occurring in φ. If L is a class of formulas, then $\mathrm{L}[\tau]$ denotes the class of all $\varphi \in \mathrm{L}$ with $\tau(\varphi) \subseteq \tau$. FO denotes the class of all first-order formulas, and EFO the subclass of all *existential* first-order formulas, i.e. all formulas that do not contain any universal quantifiers and in which existential quantifiers only occur in the scope of an even number of negation symbols.
The set of all atomic subformulas of a formula φ is denoted by $\mathrm{at}(\varphi)$. A *free variable* of a first-order formula is a variable x not in the scope of a quantifier $\exists x$ or $\forall x$. The set of all free variables of a formula φ is denoted by $\mathrm{free}(\varphi)$, the set of all variables of φ by $\mathrm{var}(\varphi)$. The notation $\varphi(x_1, \dots, x_k)$ indicates that $\mathrm{free}(\varphi) = \{x_1, \dots, x_k\}$. A *sentence* is a formula without free variables.
For a formula $\varphi(x_1, \dots, x_k)$, a $\tau(\varphi)$-structure $\mathcal{A}$, and elements $a_1, \dots, a_k \in A$, we write $\mathcal{A} \models \varphi(a_1, \dots, a_k)$ to say that $\mathcal{A}$ satisfies φ if the variables $x_1, \dots, x_k$ are interpreted by the elements $a_1, \dots, a_k$, respectively. We let

$$\varphi(\mathcal{A}) := \{(a_1, \dots, a_k) \in A^k \mid \mathcal{A} \models \varphi(a_1, \dots, a_k)\}.$$

We shall study the complexity of the following problem for fragments $\mathrm{L} \subseteq \mathrm{FO}$:

L-EVALUATION	
Input:	Formula $\varphi \in \mathrm{L}$, $\tau(\varphi)$-structure $\mathcal{A}$.
Problem:	Decide if $\varphi(\mathcal{A}) \neq \emptyset$.

Note that we have turned the actual problem of evaluating the input query, i.e. computing the set $\varphi(\mathcal{A})$, into a decision problem here. This is appropriate for the complexity theoretic considerations made in this paper.
The natural parameterization of query-evaluation is by the length of the input formula:

PARAMETERIZED L-EVALUATION	
Input:	Formula $\varphi \in \mathrm{L}$, $\tau(\varphi)$-structure $\mathcal{A}$.
Parameter:	$\|\|\varphi\|\|$
Problem:	Decide if $\varphi(\mathcal{A}) \neq \emptyset$.

FO-EVALUATION is PSPACE-complete [13], and PARAMETERIZED FO-EVALUATION is complete for the parameterized complexity class AW[1] [5]that it is W[1]-hard.

Conjunctive Queries. A *conjunctive query* is a first-order formula of the form $\exists x_1 \dots \exists x_k (\alpha_1 \wedge \dots \wedge \alpha_n)$, where $\alpha_1, \dots, \alpha_n$ are atoms. The class of all conjunctive queries is denoted by CQ.
CQ-EVALUATION is NP-complete [1], and PARAMETERIZED CQ-EVALUATION is W[1]-complete [11]. Both hardness results are immediate consequences of the simple observation that for every k there is a conjunctive query of length $O(k^2)$ saying that a graph has a k-clique. Actually, it is easy to see that CQ-EVALUATION and HOMOMORPHISM are basically the same problem.

The graph $\mathcal{G}(\varphi)$ of a first-order formula φ has vertex set $G(\varphi) := \mathrm{var}(\varphi)$ and edge set

$$E^{\mathcal{G}(\varphi)} := \big\{(x, y) \mid x \neq y, \exists \alpha \in \mathrm{at}(\varphi) : \ x, y \in \mathrm{var}(\alpha)\big\}$$

For a class C of graphs we let $\mathrm{CQ(C)} := \{\gamma \in \mathrm{CQ} \mid \mathcal{G}(\gamma) \in \mathrm{C}\}$.

Theorem 14 (Chekuri and Rajaraman [2]). *Let* C *be a class of graphs of bounded tree-width. Then* CQ(C)-EVALUATION *is in* PTIME.

Remark 2. Chekuri and Rajaraman define the graph of a formula differently; they take the incidence graph between atoms and variables. Consequently, they obtain a different notion of tree-width. But this difference is irrelevant here. Our version of the theorem has first been stated by Kolaitis and Vardi [10].
We refer the reader to [6] for a discussion of the different concepts.

Theorem 15. *Let* C *be a class of graphs of unbounded tree-width that is closed under taking minors and* τ *a vocabulary that is at least binary. Then* CQ(C)[τ]-EVALUATION *is* NP*-complete.*

Proof: Without loss of generality we can assume that $\tau = \{E\}$ for the binary relation symbol E. Observe that for every directed (n, n)-grid $\mathcal{G}$ there is a conjunctive query $\gamma_{\mathcal{G}}$ with $\mathcal{G}(\gamma_{\mathcal{G}}) = \mathcal{G}_{n,n}$ such that for every directed graph $\mathcal{D}$ we have

$$\mathcal{D} \models \gamma_{\mathcal{G}} \iff \mathrm{HOM}(\mathcal{G}, \mathcal{D}) \neq \emptyset.$$

Since C is closed under taking minors, by Robertson and Seymour's Excluded Grid Theorem 4 it contains all grids, thus for every directed square grid $\mathcal{S}$ the query $\gamma_{\mathcal{S}}$ belongs to CQ(C). Thus we can reduce DIRECTED GRID HOMOMORPHISM to CQ(C)[τ]-EVALUATION. Then the statement of the theorem follows from the NP-completeness of directed grid homomorphism (Theorem 13). $\square$

Corollary 16. *Assume that* PTIME $\neq$ NP. *Let* C *be a class of graphs that is closed under taking minors and* τ *a vocabulary that contains at least one binary relation symbol. Then the following three statements are equivalent:*

(1) CQ(C)-EVALUATION *is in* PTIME.

(2) CQ(C)[τ]-EVALUATION *is in* PTIME.

(3) C *has bounded tree-width.*

With a little extra work, on the parameterized level we get an even cleaner picture; we can drop the restriction of being closed under taking minors that we put on the class C of underlying graphs in Theorem 15.

Theorem 17. *Let* C *be a class of graphs of unbounded tree-width and* τ *a vocabulary that is at least binary. Then* PARAMETERIZED CQ(C)[τ]-EVALUATION *is* W[1]*-complete under parameterized T-reductions.*

In the proof, we will use the following lemma which is an extension of the second part of Theorem 13.

Lemma 18. *Let* C *be a class of graphs such that for every* $n \geq 1$ *there is a graph* $\mathcal{G} \in \mathrm{C}$ *with* $G = G_{n,n}$ *and* $E^{\mathcal{G}} \supseteq E^{\mathcal{G}_{n,n}}$. *Then the following problem is* W[1]*-complete under parameterized T-reductions:*

PARAMETERIZED DIRECTED-C-HOMOMORPHISM	
Input:	Directed graph $\mathcal{A}$ whose underlying graph is in C, directed graph $\mathcal{B}$.
Parameter:	$\lVert\mathcal{A}\rVert$
Problem:	Decide if there is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$.

Proof: Let us first assume that C is computable. We shall prove that

PARAMETERIZED CLIQUE

$\leq^{\mathrm{fp}}_{\mathrm{m}}$ PARAMETERIZED DIRECTED C-HOMOMORPHISM.

This even shows that PARAMETERIZED DIRECTED-C-HOMOMORPHISM is W[1]-complete under parameterized m-reductions (for computable C). For every graph $\mathcal{B}$ and natural number k we define directed graphs $\mathcal{A}^*_k$ and $\mathcal{B}^*_k$ such that the underlying graph of $\mathcal{A}^*_k$ belongs to C and there is a homomorphism from $\mathcal{A}^*_k$ to $\mathcal{B}^*_k$ if, and only if, $\mathcal{B}$ has a k-clique. Furthermore, we will do this in such a way that $\mathcal{A}^*_k$ only depends on k, the mapping $k \mapsto \mathcal{A}^*_k$ is computable, and $\mathcal{B}^*_k$ can be computed from $(\mathcal{B}, \mathcal{A}^*_k)$ in polynomial time.
So let $k \geq 1$ and $K := \binom{k}{2}$. Let $\mathcal{G} \in \mathrm{C}$ such that $G = G_{K,K}$ and $E^{\mathcal{G}} \supseteq E^{\mathcal{G}_{K,K}}$. Since C is computable, such a $\mathcal{G}$ can be found effectively. Now we let $\mathcal{A}^*_k$ be a directed graph with underlying graph $\mathcal{G}$ such that $\mathrm{HOM}(\mathcal{A}^*_k, \mathcal{A}^*_k) = \{\mathrm{id}\}$. Such an $\mathcal{A}^*_k$ exists by Lemma 12. Let $\mathcal{B}$ be an arbitrary graph. Recall that f is a bijection from $\{1, \ldots, K\} \to \{\{i, j\} \mid 1 \leq i < j \leq k\}$ that maps l to the lth element in the lexicographic order of $\big\{\{i, j\} \bigm| 1 \leq i < j \leq k\big\}$.
We define the directed graph $\mathcal{B}^*_k$ as follows:

$$\begin{aligned}
B^*_k :=& \Big\{(a, e, i, p) \Bigm| a \in B, e \in E^{\mathcal{B}}, 1 \leq i \leq k, 1 \leq p \leq K \\
&\qquad \text{such that } \big(i \in f(p) \iff a \in e\big)\Big\} \\
&\cup \big\{(i, p) \mid k + 1 \leq i \leq K, 1 \leq p \leq K\big\}, \\
E^{\mathcal{B}^*_k} :=& \Big\{\big((a, e, i, p), (a', e', i', p')\big) \Bigm| a, a' \in B, e, e' \in E^{\mathcal{B}}, \\
&\qquad 1 \leq i, i' \leq k, 1 \leq p, p' \leq K, \\
&\qquad \big((i, p), (i', p')\big) \in E^{\mathcal{A}^*_k}, \\
&\qquad \big(i = i' \implies a = a'\big), \big(p = p' \implies e = e'\big)\Big\} \\
&\cup \Big\{\big((a, e, i, p), (i', p')\big) \Bigm| a \in B, e \in E^{\mathcal{B}}, \\
&\qquad 1 \leq i \leq k, k + 1 \leq i' \leq K, 1 \leq p, p' \leq K, \\
&\qquad \big((i, p), (i', p')\big) \in E^{\mathcal{A}^*_k}\Big\} \\
&\cup \Big\{\big((i, p), (a, e, i', p'),\big) \Bigm| a \in B, e \in E^{\mathcal{B}}, \\
&\qquad 1 \leq i' \leq k, k + 1 \leq i \leq K, 1 \leq p, p' \leq K, \\
&\qquad \big((i, p), (i', p')\big) \in E^{\mathcal{A}^*_k}\Big\} \\
&\cup \Big\{\big((i, p), (i', p')\big) \Bigm| k + 1 \leq i, i' \leq K, 1 \leq p, p' \leq K, \\
&\qquad \big((i, p), (i', p')\big) \in E^{\mathcal{A}^*_k}\Big\}.
\end{aligned}$$

Then it can be proved analogously to Lemma 9 that $\mathcal{B}$ contains a k-clique if, and only if, there is a homomorphism $h : \mathcal{A}_k^* \to \mathcal{B}_k^*$. Moreover, it is easy to see that, given $\mathcal{B}$ and $\mathcal{A}_k^*$, $\mathcal{B}_k^*$ can be computed in polynomial time.

It remains to deal with non-computable classes C. For such classes, we can only prove that

$$\text{PARAMETERIZED CLIQUE}$$
$$\leq_{\mathrm{T}}^{\mathrm{fp}} \text{PARAMETERIZED DIRECTED-C-HOMOMORPHISM}.$$

The only place in the reduction above where we used our assumption that C is computable was when for a given K we computed the graph $\mathcal{G} \in \mathrm{C}$ with $G = G_{K,K}$ and $E^{\mathcal{G}} \supseteq E^{\mathcal{G}_{K,K}}$. Now we can no longer compute such a $\mathcal{G}$, we merely know that it exists. To circumvent this problem, we simply do the reduction described above for every graph $\mathcal{G}$ with $G = G_{K,K}$ and $E^{\mathcal{G}} \supseteq E^{\mathcal{G}_{K,K}}$.
Let us explain the procedure in detail: We are given a k and a directed graph $\mathcal{B}$ and want to decide whether $\mathcal{B}$ contains a k-clique, using an oracle for PARAMETERIZED DIRECTED-C-HOMOMORPHISM. As above, we let $K := \binom{k}{2}$. For every graph $\mathcal{G}$ with $G = G_{K,K}$ and $E^{\mathcal{G}} \supseteq E^{\mathcal{G}_{K,K}}$ we do the following: As described above, we compute directed graphs $\mathcal{A}_k^*$ and $\mathcal{B}_k^*$ such that the underlying graph of $\mathcal{A}_k^*$ is $\mathcal{G}$ and there is a homomorphism from $\mathcal{A}_k^*$ to $\mathcal{B}_k^*$ if, and only if, $\mathcal{B}$ has a k-clique. Then we use our oracle and let it decide if there is a homomorphism from $\mathcal{A}_k^*$ to $\mathcal{B}_k^*$. The problem is that the answer we get may be wrong if the graph $\mathcal{G}$ we started with is not contained in C.
We distinguish between two cases. If the oracle answers 'no' for all $\mathcal{G}$, in particular it answers 'no' for all $\mathcal{G}$ in C (remember that at least one such $\mathcal{G}$ exists). Therefore, we know that the answer is correct. If we get the answer 'yes' for some $\mathcal{G}$, we can try to compute a homomorphism from $\mathcal{A}_k^*$ to $\mathcal{B}_k^*$ using Lemma 2. Then we can easily check if the answer was correct. If we find a correct 'yes'-instance, we know that $\mathcal{B}$ contains a k-clique. Otherwise, we know that for all $\mathcal{G}$ in C our oracle answers 'no', and thus that $\mathcal{B}$ does not contain a k-clique. □

We are now ready to prove Theorem 17

Proof: Again, without loss of generality we assume that $\tau = \{E\}$. By the Excluded Grid Theorem 4, for every $n \geq 1$ there exists a graph $\mathcal{G}_n \in \mathrm{C}$ such that $\mathcal{G}_{n,n} \preceq \mathcal{G}_n$. We let $\mathcal{G}_n'$ be a connected component of $\mathcal{G}_n$ such that $\mathcal{G}_{n,n} \preceq \mathcal{G}_n'$ and $\mu_n : \mathcal{G}_{n,n} \preceq \mathcal{G}_n'$ a minor map such that $\bigcup_{a \in G_{n,n}} \mu_n(a) = G_n'$. We can always find such a minor map because $\mathcal{G}_n'$ is connected. We let $\mathcal{G}_n''$ be the graph with universe $G_n'' := G_{n,n}$ and edge set

$$E^{\mathcal{G}_n''} := \big\{(a,b) \bigm| \exists a' \in \mu_n(a), b' \in \mu_n(b) :\ (a',b') \in E^{\mathcal{G}_n'}\big\}.$$

Then $E^{\mathcal{G}_n''} \supseteq E^{\mathcal{G}_{n,n}}$. We let $\mathrm{C}'' := \{\mathcal{G}_n'' \mid n \geq 1\}$. Recall Lemma 18. We shall prove that

$$\text{PARAMETERIZED DIRECTED-C}''\text{-HOMOMORPHISM}$$
$$\leq_{\mathrm{m}}^{\mathrm{fp}} \text{PARAMETERIZED CQ(C)}[\tau]\text{-EVALUATION}.$$

For every $n \geq 1$, we define a conjunctive query $\gamma_n \in \mathrm{CQ(C)}$ such that for every directed graph $\mathcal{D}$ we have:

$$\mathrm{HOM}(\mathcal{G}_n'', \mathcal{D}) \neq \emptyset \iff \mathcal{D} \models \gamma_n. \tag{2}$$

Let $\mathcal{G}_n \in \mathrm{C}$ be the graph, $\mathcal{G}_n' \subseteq \mathcal{G}_n$ its connected component, and $\mu_n : \mathcal{G}_{n,n} \preceq \mathcal{G}_n'$ the minor map used to define $\mathcal{G}_n''$.

We choose a variable x_b for every $b \in G_n$ (such that $x_b \neq x_{b'}$ for $b \neq b'$) and define three sets of atoms:

$$A_1 := \{x_b = x_{b'} \mid (b,b') \in E^{\mathcal{G}_n}, \exists a \in G_{n,n} :\ b,b' \in \mu_n(a)\}$$
$$A_2 := \Big\{Ex_bx_{b'} \Bigm| (b,b') \in E^{\mathcal{G}_n}, \exists (a,a') \in E^{\mathcal{G}_n''} : \big(b \in \mu_n(a), b' \in \mu_n(a')\big)\Big\}$$
$$A_3 := \big\{x_b = x_{b'} \mid (b,b') \in E^{\mathcal{G}_n}, b,b' \notin G_n'\big\}.$$

Let $\bar{x}$ be a tuple that contains all the x_b, $A := A_1 \cup A_2 \cup A_3$ and $\gamma_n := \exists \bar{x} \bigwedge_{\alpha \in A} \alpha$.
Note first that $\mathcal{G}(\gamma_n) = \mathcal{G}_n$, thus $\gamma_n \in \mathrm{CQ(C)}$. Now let $\mathcal{D}$ be a directed graph. To prove the forward direction of (2), assume that $h : \mathcal{G}_n'' \to \mathcal{D}$ is a homomorphism. Then to see that $\mathcal{D} \models \gamma_n$, for every $a \in G_n''$ we interpret all variables x_b with $b \in \mu_n(a)$ by $h(a)$. For every connected component C of $G_n \setminus G_n'$, we pick one element c and interpret all x_b with $b \in C$ by c. Then all atoms in A_1 and A_3 are satisfied, and it can be easily checked that this is also the case for the atoms in A_2.
Conversely, suppose that $\mathcal{D} \models \gamma_n$. Then by A_1 and because $\mu_n(a)$ is connected in $\mathcal{G}_n$, for every $a \in G_n''$ there is a unique element $d \in D$ that interprets all the x_b with $b \in \mu_n(a)$. We let $h(a) := d$, then A_2 guarantees that this gives us a homomorphism $h : \mathcal{G}_n'' \to \mathcal{D}$. □

Corollary 19. *Assume that* FPT $\neq$ W[1]. *Let* C *be an arbitrary class of graphs and* τ *a vocabulary that contains at least one binary relation symbol. Then the following statements are equivalent:*

(1) PARAMETERIZED CQ(C)-EVALUATION *is in* FPT.

(2) PARAMETERIZED CQ(C)[τ]-EVALUATION *is in* FPT.

(3) CQ(C)-EVALUATION *is in* PTIME.

(4) CQ(C)[τ]-EVALUATION *is in* PTIME.

(5) C *has bounded tree-width.*

Proof: The implications (3) $\Rightarrow$ (4), (3) $\Rightarrow$ (1), (4) $\Rightarrow$ (2), and (1) $\Rightarrow$ (2) are trivial. (5) $\Rightarrow$ (3) by Theorem 14, and (2) $\Rightarrow$ (5) by Theorem 17. □

Remark 3. In the proof of Theorem 17 we used the fact that equality, =, was part of the vocabulary. We dont know yet whether the theorem remains true without this requirement.

Remark 4. Yannakakis [15] proved that the evaluation of *acyclic* conjunctive queries is possible in polynomial time (and therefore tractable both in the classical sense of combined complexity and fixed-parameter tractable). This does not contradict Corollaries 19 and 16 because the class of acyclic conjunctive queries is *not* of the form CQ(C) for a class of graphs. However, the underlying graph of a conjunctive query whose vocabulary contains at most k-ary relation symbols, for a $k \geq 2$, has tree-width at most $k-1$. Thus in particular, if the vocabulary is fixed, acyclic conjunctive queries have tree-width bounded by the arity of the vocabulary.

Remark 5. A *conjunctive query with negation* is a formula of the form $\exists x_1 \ldots \exists x_k (\lambda_1 \wedge \ldots \wedge \lambda_n)$, where the λ_i are atomic or negated atomic formulas. It is easy to see that our results extend to conjunctive queries with negation.

5. Towards First-Order Evaluation

Definition 3. A fragment $\mathrm{L} \subseteq \mathrm{FO}$ is *closed* if it satisfies the following condition: If $\varphi \in \mathrm{L}$, α is an atom in φ, and β a quantifier-free formula with $\mathrm{var}(\alpha) = \mathrm{var}(\beta)$, then the formula φ' obtained from φ by replacing α by β is also contained in L.

The motivation for this definition is that closed classes of first-order formuls somehow resemble classes of conjuctive queries of the form CQ(C) for a class C of graphs. There is no direct formal correspondence between the two notions, though, because we have to admit negated atoms here.)

Let $\mathrm{L}_1, \mathrm{L}_2 \subseteq \mathrm{FO}$. We say that L_1 is *effectively contained* in L_2 (and write $\mathrm{L}_1 \leq_{\mathrm{eff}} \mathrm{L}_2$) if there is an algorithm that computes, given a formula $\varphi_1 \in \mathrm{L}_1$, an equivalent formula $\varphi_2 \in \mathrm{L}_2$.

Theorem 20. *Let* L *be a closed class of* EFO*-sentences that is recursively enumerable. Assume that* $\mathrm{FPT} \neq \mathrm{W}[1]$*. Then* PARAMETERIZED L-EVALUATION *is in* FPT *if, and only if, there exists a* $k \geq 1$ *such that* $\mathrm{L} \leq_{\mathrm{eff}} \mathrm{EFO}^k$.

Proof: The backward direction is well-known.
To prove the forward direction, we assume that PARAMETERIZED L-EVALUATION is in FPT.
Let $\varphi \in \mathrm{L}$. By standard techniques we transform φ into a sentence φ' in prenex normal form in which negation symbols only occur in front of atomic subformulas. In doing so, we may have to rename some of the variables, but we do not have to duplicate any atoms. In other words, there is a canonical way to associate an atom α' of φ' with every atom α of φ.
Now we bring the quantifier-free part of φ' into disjunctive normal form. Switching existential quantifiers and disjunctions, we obtain a formula φ'' of the form $\bigvee_{i\in I} \gamma_i^\varphi$, where the γ_i^φ are conjunctive queries with negation. If we do this in the straightforward way, φ' and φ'' will have precisely the same set of atoms, but every atom of φ' may occur several times in φ''.
Consider the class C of all graph $\mathcal{G}(\gamma_i^\varphi)$, for all $\varphi \in \mathrm{L}$. We claim that there is a parameterized m-reduction from PARAMETERIZED CQ(C)[$\{E\}$]-EVALUATION to PARAMETERIZED L-EVALUATION. To see this, let $\gamma \in \mathrm{CQ(C)}[\{E\}]$. Then there is a formula $\varphi \in \mathrm{L}$ such that for some γ_i^φ occurring in φ'' we have $\mathcal{G}(\gamma_i^\varphi) = \mathcal{G}(\gamma)$. Because all the literals involve only binary relations, it is easy to see that we can replace the literals of γ_i^φ in a suitable way by a conjunction of atoms such that the resulting formula is γ; equivalently we can replace the atoms by quantifier-free formulas. We do the corresponding substitution in φ' and replace all other literals of φ' by a quantifier free formula that is not satisfiable (for example, a conjunction of atoms $x \neq x$). Thus in φ' every γ_j^φ with $j \neq i$ is replaced by an unsatisfiable formula and γ_i^φ is replaced by γ. Therefore the resulting formula γ' is equivalent to γ. Now we go back from φ' to φ. We get a formula γ'' equivalent to γ that can be obtained from φ by replacing atoms by quantifier-free formulas with the same variables. Since the fragment L is closed, γ'' is contained in L. Thus for every $\gamma \in \mathrm{CQ(C)}[\{E\}]$ there is an equivalent $\gamma'' \in \mathrm{L}$, and since L is recursively enumerable and equivalence of conjunctive queries is decidable, such a γ'' can actually be computed. This proves our claim.
Since we assumed PARAMETERIZED L-EVALUATION to be fixed-parameter tractable, by Theorem 19 the class C has bounded tree-width, say w. Now we apply a result of Kolaitis and Vardi [10] saying that every conjunctive query of tree-width at most w can be effectively translated into an equivalent EFO^{w+1}-formula. (Actually, we use an extension of this theorem for conjunctive queries with negation.) Thus all the γ_i^φ are equivalent to EFO^{w+1}-sentences. This implies that φ'' and thus φ is equivalent to an EFO^{w+1}-formula. Furthermore, our proof shows that the translation is effective. □

Remark 6. The previous theorem can be extended to formulas with at most k free variables, but is clearly wrong for formulas with more than k free variables.

Conjecture 21. *Assume that* $\mathrm{FPT} \neq \mathrm{W}[1]$*. Let* $\mathrm{L} \subseteq \mathrm{FO}$ *be closed and recursively enumerable. Then* PARAMETERIZED L-EVALUATION *is in* FPT *if, and only if, there exists a* $k \geq 1$ *such that* $\mathrm{L} \leq_{\mathrm{eff}} \mathrm{FO}^k$.

The idea to prove the conjecture would be to use an analogue of Kolaitis and Vardi's results for FO, which is contained in [6] (Theorem 5.29).

6. Conclusions

The results of this paper give a good understanding on when the evaluation of conjunctive queries becomes a tractable problem. The main contribution is to show that evaluating conjunctive queries is tractable for those queries that have a bounded tree-width structure (under the hypothesis that NP≠ PTIME and FPT≠ W[1]). Beyond conjunctive queries, it was known that for each k, FO^k could be evaluated in PTIME and in FPT. The second contribution of the paper is to propose a converse of this: Tractability of query evaluation implies an expressive power limited by the number of variables. This is formalized in conjecture 21, which we could prove only for the existential fragment of FO. A third contribution of independent interest, is that *colored grid homomorphism* problem is NP-complete and, if parameterized by the grid size, W[1]-complete.

Acknowledgments

We thank Jörg Flum, Georg Gottlob and Victor Vianu for helpful discussions and comments.

7. References

[1] A.K. Chandra and P.M. Merlin. Optimal implementation of conjunctive queries in relational data bases. In *Proceedings of the 9th ACM Symposium on Theory of Computing*, pages 77–90, 1977.

[2] Ch. Chekuri and A. Rajaraman. Conjunctive query containment revisited. In Ph. Kolaitis and F. Afrati, editors, *Proceedings of the 5th International Conference on Database Theory*, volume 1186 of *Lecture Notes in Computer Science*, pages 56–70. Springer-Verlag, 1997.

[3] R.G. Downey and M.R. Fellows. Fixed-parameter tractability and completeness II: On completeness for $W[1]$. *Theoretical Computer Science*, 141:109–131, 1995.

[4] R.G. Downey and M.R. Fellows. *Parameterized Complexity*. Springer-Verlag, 1999.

[5] R.G. Downey, M.R. Fellows, and U. Taylor. The parameterized complexity of relational database queries and an improved characterization of $W[1]$. In Bridges, Calude, Gibbons, Reeves, and Witten, editors, *Combinatorics, Complexity, and Logic – Proceedings of DMTCS '96*, pages 194–213. Springer-Verlag, 1996.

[6] J. Flum, M. Frick, and M. Grohe. Query evaluation via tree-decompositions. In J. van den Bussche and V. Vianu, editors, *Proceedings of the 8th International Conference on Database Theory*, volume 1973 of *Lecture Notes in Computer Science*, pages 22–38. Springer Verlag, 2001.

[7] G. Gottlob, N. Leone, and F. Scarcello. The complexity of acyclic conjunctive queries. In *Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, pages 706–715, 1998.

[8] G. Gottlob, N. Leone, and F. Scarcello. Hypertree decompositions and tractable queries. In *Proceedings of the 18th ACM Symposium on Principles of Database Systems*, pages 21–32, 1999.

[9] R.M. Karp. Reducibilities among combinatorial problems. In R.E. Miller and J.W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, New York, 1972.

[10] Ph.G. Kolaitis and M.Y. Vardi. Conjunctive-query containment and constraint satisfaction. In *Proceedings of the 17th ACM Symposium on Principles of Database Systems*, pages 205–213, 1998.

[11] C.H. Papadimitriou and M. Yannakakis. On the complexity of database queries. In *Proceedings of the 16th ACM Symposium on Principles of Database Systems*, pages 12–19, 1997.

[12] N. Robertson and P.D. Seymour. Graph minors V. Excluding a planar graph. *Journal of Combinatorial Theory, Series B*, 41:92–114, 1986.

[13] L.J. Stockmeyer. *The Complexity of Decision Problems in Automata Theory*. PhD thesis, Department of Electrical Engineering, MIT, 1974.

[14] M. Y. Vardi. The complexity of relational query languages. In *Proceedings of the 14th ACM Symposium on Theory of Computing*, pages 137–146, 1982.

[15] M. Yannakakis. Algorithms for acyclic database schemes. In *7th International Conference on Very Large Data Bases*, pages 82–94, 1981.

[16] M. Yannakakis. Perspectives on database theory. In *Proceedings of the 36th Annual IEEE Symposium on Foundations of Computer Science*, pages 224–246, 1995.