Characterising Fixed Parameter Tractability for Query Evaluation over Guarded TGDs

Cristina Feier ✉
University of Bremen, Germany

Abstract

We consider the parameterized complexity of evaluating Ontology Mediated Queries (OMQ) based on Guarded TGDs (GTGD) and Unions of Conjunctive Queries, in the case where relational symbols have unrestricted arity and where the parameter is the size of the OMQ. We establish exact criteria for fixed-parameter tractable (fpt) evaluation of recursively enumerable (r.e.) classes of such OMQs (under the widely held Exponential Time Hypothesis). One of the main technical tools introduced in the paper is an fpt-reduction from deciding parameterized uniform CSPs to parameterized OMQ evaluation. The reduction preserves measures known to be essential for classifying r.e. classes of parameterized uniform CSPs: submodular width (according to the well known result of Marx for unrestricted-arity schemas) and treewidth (according to the well known result of Grohe for bounded-arity schemas). As such, it can be employed to obtain hardness results for evaluation of r.e. classes of parameterized OMQs based on GTGD both in the unrestricted and in the bounded arity case. Previously, for bounded arity schemas, this has been tackled using a technique requiring full introspection into the construction employed by Grohe.

2012 ACM Subject Classification Theory of computation → Database theory

Keywords and phrases omq, fpt evaluation, guarded tgds, unbounded arity, submodular width

Digital Object Identifier 10.4230/LIPIcs.ICDT.2022.12

Related Version *Full Version*: `https://arxiv.org/abs/2101.11727`

Acknowledgements I would like to thank the reviewers for the useful comments.

1 Introduction

Ontology mediated querying refers to the scenario where queries are posed to a database enhanced with a logical theory, commonly referred to as an *ontology*. The ontology refines the specific knowledge provided by the database by means of a logical theory. Popular ontology languages are decidable fragments of first order logic (FOL) like description logics [1], GTGD [11], or monadic disjunctive datalog [8]. Non-monotonic formalisms like answer set programming [28], or combinations of languages from the former and latter category like r-hybrid knowledge bases [31], g-hybrid knowledge bases [25], etc. have also been considered. As concerns query languages, atomic queries (AQs), conjunctive queries (CQs), and unions thereof (UCQs) are commonly used. An OMQ language is a tuple $(\mathcal{L}, \mathcal{Q})$, where $\mathcal{L}$ is an ontology language, and $\mathcal{Q}$ is a query language.

One thoroughly explored fragment of FOL as a basis for ontology specification languages is that of *tuple generating dependencies* (TGD). A tgd is a rule (logical implication) having as body and head conjunctions of atoms, where some variables occurring in head atoms might be existentially quantified (all other variables are universally quantified). As such, it potentially allows the derivation of atoms over fresh individuals (individuals not mentioned in the database). Answering even AQs with respect to sets of tgds is undecidable [11]. There have been lots of work on identifying decidable fragments [11, 2, 13]. A prominent such fragment is guarded TGDs (GTGD) [11]: a tgd is *guarded* if all universally quantified variables occur as terms of some body atom, called *guard*.

While query answering with respect to GTGD is decidable, the combined complexity of the problem is quite high: EXPTIME-complete for bounded arity schemas, and 2EXPTIME-complete in general. A natural question is when can OMQs from (GTGD, UCQ) be evaluated efficiently? A first observation is that by fixing the set of tgds, the complexity drops to NP for evaluating CQs, and to PTIME for evaluating AQs and CQs of bounded treewidth [12], which is similar to the complexity of query evaluation over databases [32].

Efficiency of query evaluation over databases has been a long evolving topic: starting with results concerning tractability of acyclic CQs evaluation [32], extended to bounded treewidth CQs in [15], and culminating, in the case of bounded arity schemas, with the result of Grohe which characterizes classes of CQs which can be efficiently evaluated in a parameterized complexity framework where the parameter is the query size. In this setting, under the assumption that FPT $\neq$ W[1], Grohe [24] establishes that exactly those r.e. classes of CQs which have bounded treewidth modulo homomorphic equivalence are fixed-parameter tractable. In this case fpt also coincides with polytime evaluation.

As concerns OMQs from (GTGD, UCQ) over bounded arity schemas, a similar characterization to that of Grohe has been established in a parameterized setting where the parameter is the size of the OMQ [3]. The cut-off criterium for efficient evaluation is again bounded treewidth modulo equivalence, only this time equivalence takes into account also the ontology. An OMQ from (GTGD, UCQ) has *semantic treewidth* k if there exists an equivalent OMQ from (GTGD, UCQ) whose UCQ has treewidth k [5]. Then, under the assumption that FPT $\neq$ W[1], a r.e. class of OMQs from (GTGD, UCQ) over bounded arity schemas can be evaluated in FPT iff it has bounded semantic treewidth. The similarity of the characterization with the database case is not coincidental: the results for OMQs build on the results of Grohe in a non-trivial way. In fact, the lower bound proof uses a central construction from [24], but has to employ sophisticated techniques to adapt this to OMQs.

The main open question which is adressed in this paper is: *when is it possible to efficiently evaluate OMQs from (GTGD, UCQ) in the general case, i.e. when there is no restriction concerning schema arity?* We again consider a parameterized setting, where the parameter is the size of the OMQ. This is a reasonable choice as the size of the OMQ is usually much smaller than the size of the database. As such, we are interested in *investigating the limits of fixed-parameter tractability of evaluating a class of OMQs $\mathbb{Q}$ from (GTGD, UCQ).* We denote such a parameterized problem as p-OMQ($\mathbb{Q}$).

Before stating our main results, we review some results concerning efficiency of solving *constraint satisfaction problems* (CSP). These are relevant as CQ evaluation over databases is tightly linked to solving so-called *uniform CSPs.* Given classes of relational structures $\mathbb{A}$ and $\mathbb{B}$, a CSP problem $(\mathbb{A}, \mathbb{B})$ asks whether there exists a homomorphism from some relational structure in $\mathbb{A}$ to another relational structure in $\mathbb{B}$. In the uniform case, $\mathbb{A}$ is fixed, $\mathbb{B}$ is the class of all relational structures and the problem is denoted as $(\mathbb{A}, \_)$. The parameterized version of the problem (where for a problem instance (A, B) the parameter is the size of A) is denoted as p-CSP$(\mathbb{A}, _)$. When restricted to classes of finite structures, uniform CSPs can be seen as an alternative presentation of the problem of evaluating a class of Boolean CQs over databases. In fact, Grohe's characterization for fpt evaluation of r.e. classes of CQs in the bounded arity case has been achieved via a uniform CSP detour [24].

In the unrestricted arity case, Marx [30] established in a seminal result the border for fpt evaluation of uniform CSPs of the form p-CSP$(\mathbb{A}, _)$, where $\mathbb{A}$ is closed under underlying hypergraphs. The restriction has been lifted in [16], yielding a full characterization for parameterized uniform CSPs of unrestricted arity. Both results are based on a widely held conjecture, the Exponential Time Hypothesis [26] and rely on a new structural measure, *submodular width.* The measure will also play a major role in our characterizations:

Main Result 1. Let $\mathbb{Q}$ be a r.e. class of OMQs from (GTGD, UCQ). Assuming the Exponential Time Hypothesis, p-OMQ($\mathbb{Q}$) is fixed-parameter tractable iff $\mathbb{Q}$ has bounded semantic submodular width.

To prove the result, we exploit the fact that every OMQ from (GTGD, UCQ) can be rewritten into an OMQ from (GDLog, UCQ) [3], where GDLog stands for Guarded Datalog, the restriction of GTGD to rules with only universally quantified variables. For OMQs from (GDLog, UCQ), we construct equivalent OMQs called *covers* which are witnesses for bounded semantic submodular width. Covers are based on sets of *characteristic databases for OMQs* which are databases that entail the OMQs and which are small with respect to the homomorphism order, in a very specific sense. While in a database setting, the database induced by a CQ can be seen as a canonical database which entails the query, in the case of OMQs which pose restrictions on the database schema this might not be possible: a CQ in an OMQ might contain symbols which are not allowed to occur in a database. In fact, this is a typical usage of ontologies: to enrich the database schema with new terminology. As such, we identify other representative databases for OMQs. Based on these notions, for OMQs from (GDLog, UCQ), we establish the following:

Main Result 2. For $\mathbb{Q}$ a r.e. class of OMQs from (GDLog, UCQ), let $\mathbb{Q}_c$ and $\mathbb{D}_{\mathbb{Q}}$ be the classes of covers and characteristic databases for OMQs from $\mathbb{Q}$, respectively. Under the Exponential Time Hypothesis, the following statements are equivalent:

1. p-OMQ($\mathbb{Q}$) is fixed-parameter tractable;
2. $\mathbb{Q}_c$ has bounded submodular width;
3. $\mathbb{D}_{\mathbb{Q}}$ has bounded submodular width.

The hardness result for the above characterization is obtained via an fpt-reduction from parameterized uniform CSP evaluation to parameterized OMQ evaluation.

Main Result 3. For $\mathbb{Q}$ a r.e. class of OMQs from (GDLog, UCQ), there exists an fpt-reduction from p-CSP($\mathbb{D}_{\mathbb{Q}}, \_$) to p-OMQ($\mathbb{Q}$).

The reduction is important as a stand-alone result as it enables lifting of other hardness complexity results from the database world to the OMQ one. It enables us to obtain analogues of our Main Result 1 and Main Result 2 for the case of OMQs based on GTGD with bounded arity schemas using the results from [24] as a black-box. In this case, submodular width is replaced with treewidth and the Exponential Time Hypothesis with the assumption that FPT $\neq$ W[1]. As such, we re-establish the semantic characterization from [3], and we provide an alternative characterization for fpt evaluation of OMQs from (GDLog, UCQ) over bounded arity schemas.

2 Preliminaries

Structures, Databases. A *schema* $\mathbf{S}$ is a finite set of relation symbols with associated arities. An $\mathbf{S}$-*fact* has the form $r(\mathbf{a})$, where $r \in \mathbf{S}$, and $\mathbf{a}$ is a tuple of constants of size the arity of r. An $\mathbf{S}$-*structure* A is a set of $\mathbf{S}$-facts. The domain of a structure A, $\mathsf{dom}(A)$, is the set of constants which occur in facts in A. Given a structure A and a subset $C \subseteq \mathsf{dom}(A)$, the *sub-structure of A induced by C*, $A|_C$,it is the structure containing all facts $r(\mathbf{b}) \in A$ such that $\mathbf{b} \subseteq C$. *The product of two structures A and B*, $A \times B$, is a structure with domain $\mathsf{dom}(A) \times \mathsf{dom}(B)$ consisting of all facts of the form $r((a_1, b_1), \ldots, (a_n, b_n))$, where $r(a_1, \ldots, a_n) \in A$ and $r(b_1, \ldots, b_n) \in B$. Given two structures A and B, a function $f : \mathsf{dom}(A) \to \mathsf{dom}(B)$ is said to be a *homomorphism* from A to B, if for every fact $r(\mathbf{a}) \in A$, there exists a fact $r(\mathbf{b}) \in B$ such that $f(\mathbf{a}) = \mathbf{b}$. The image of A in B under f, $f(A)$, is the set of facts of the form $r(f(\mathbf{a}))$ in B, where $r(\mathbf{a})$ is from A. When such a homomorphism exists we say that A *maps into* B, denoted $A \to B$.

Two structures are *equivalent* if $A \to B$ and $B \to A$. We write $A \leftrightarrow B$. They are *isomorphic* if there exists a homomorphism h from A to B which is bijective and onto, i.e. for every fact $r(\mathbf{b}) \in B$, there exists a fact $r(\mathbf{a}) \in A$ such that $h(\mathbf{a}) = \mathbf{b}$. A structure A is a *core* if every homomorphism from A to itself is injective. Every structure A has a sub-structure A' which is equivalent to A and is a core. All cores of a structure are isomorphic. The homomorphism relation $\to$ is a pre-order over the set of all structures. When restricted to structures which are cores (taken up to isomorphism), $\to$ is a partial order. Following [21], we will refer to this order as the *homomorphism order*.

An $\mathbf{S}$-*database* is a finite $\mathbf{S}$-structure. For D a database, and $\mathbf{a} \subseteq \mathsf{dom}(D)$, $\mathbf{a}$ is a *guarded set in D* if there exists a fact $r(\mathbf{a}')$ in D such that $\mathbf{a} \subseteq \mathbf{a}'$. It is a *maximal guarded set* if there exists no guarded strict superset. As in the previous definitions, we will sometimes abuse notation by using tuples of constants to refer to the underlying sets of constants instead.

Conjunctive Queries, Atomic Queries. A *conjunctive query (CQ)* is a formula of the form $q(\mathbf{x}) = \exists \mathbf{y} \phi(\mathbf{x}, \mathbf{y})$, with $\mathbf{x}$ and $\mathbf{y}$ tuples of variables and $\phi(\mathbf{x}, \mathbf{y})$ a conjunction of atoms having as terms only variables from $\mathbf{x} \cup \mathbf{y}$. The set $\mathbf{x}$ is the set of *answer variables* of q, while the set $\mathbf{y}$ is the set of *existential variables* of q. We denote with $\mathsf{var}(q)$ the set of variables of q, and with $D[q]$ the *canonical database of* q, i.e. the set of atoms which occur in ϕ (viewed as facts). When $\mathbf{x}$ is empty, the CQ is said to be *Boolean* (BCQ). We will sometimes use BCQs or their canonical databases interchangeably. A sub-query of a BCQ q is a BCQ p such that $D[p] \subseteq D[q]$. A *union of conjunctive queries (UCQ)* is a formula of the form $q(\mathbf{x}) = q_1(\mathbf{x}) \vee \ldots \vee q_n(\mathbf{x})$, where each $q_i(\mathbf{x})$ is a CQ, for $i \in [n]$. An *atomic query (AQ)* is a CQ in which ϕ contains a single atom. In the following, whenever we refer to CQs or UCQs, we tacitly assume they are Boolean. As concerns AQs, unless stated otherwise, we assume they are of the form $r(\mathbf{x})$, i.e. they contain no existentially quantified variables.

For a structure I, a CQ $q(\mathbf{x})$, and a tuple of constants $\mathbf{a}$ from $\mathsf{dom}(I)$, $\mathbf{a}$ *is an answer to q over I*, or $I \models q(\mathbf{a})$, if there is a homomorphism h from $D[q]$ to I such that $h(\mathbf{x}) = \mathbf{a}$. If $q(\mathbf{x})$ is a UCQ of the form $q_1(\mathbf{x}) \vee \ldots \vee q_n(\mathbf{x})$, $I \models q(\mathbf{a})$ if $I \models q_i(\mathbf{a})$, for some $i \in [n]$.

Ontology Mediated Queries. An *ontology mediated query (OMQ) Q* is a triple $(\mathcal{O}, \mathbf{S}, q(\mathbf{x}))$, where $\mathcal{O}$ is an ontology, $\mathbf{S}$ is a schema, and $q(\mathbf{x})$ is a query. When $\mathcal{O}$ is specified using the ontology language $\mathcal{L}$, and q using the query language $\mathcal{Q}$, we say that Q belongs to the OMQ language $(\mathcal{L}, \mathcal{Q})$. The schema $\mathbf{S}$ specifies which relational symbols can occur in databases over which Q is evaluated. We say that Q is an $\mathbf{S}$-*OMQ*.

Given an $\mathbf{S}$-database D, and a tuple of constants $\mathbf{a}$, all of which are from $\mathsf{dom}(D)$, we say that $\mathbf{a}$ is an *answer to Q over D*, or $D \models Q(\mathbf{a})$, if $\mathcal{O} \cup D \models q(\mathbf{a})$, where $\models$ is the entailment relation in $\mathcal{L}$. For Q_1 and Q_2 two OMQs over the same schema $\mathbf{S}$, we say that Q_1 is *contained in* by Q_2, written $Q_1 \subseteq Q_2$, if for every $\mathbf{S}$-database D and tuple of constants $\mathbf{a}$: $D \models Q_1(\mathbf{a})$ implies $D \models Q_2(\mathbf{a})$. We also say that Q_1 is *equivalent* to Q_2 if $Q_1 \subseteq Q_2$ and $Q_2 \subseteq Q_1$.

TGDs, Guarded TGDs. A *tuple generating dependency* (tgd) is a first order sentence of the form $\forall \mathbf{x} \forall \mathbf{y}\ \phi(\mathbf{x}, \mathbf{y}) \to \exists \mathbf{z}\ \psi(\mathbf{x}, \mathbf{z})$ (abbreviated $\phi(\mathbf{x}, \mathbf{y}) \to \exists \mathbf{z}\ \psi(\mathbf{x}, \mathbf{z})$), with ϕ, the *body* of the tgd, and ψ, the *head* of the tgd, being conjunctions of atoms with terms from $\mathbf{x} \cup \mathbf{y}$, and from $\mathbf{x} \cup \mathbf{z}$, resp. A set of tgds is a *TGD program.* The problem of *evaluating a UCQ $q(\mathbf{x})$ over a TGD program $\mathcal{O}$ w.r.t. an* $\mathbf{S}$-*database D* consists in checking whether for some tuple

$\mathbf{a}$ over $\mathsf{dom}(D)$, it is the case that $D \models Q(\mathbf{a})$, where Q is the OMQ $(\mathcal{O}, \mathbf{S}, q(\mathbf{x}))$. While the problem is undecidable, [11], it can be characterized via a completion of the database D to a structure which is a universal model of $\mathcal{O}$ and D, called *chase* [29, 19, 27]. We will denote with $\mathsf{ch}_{\mathcal{O}}(D)$ the chase of $\mathcal{O}$ w.r.t. D. Then, $D \models Q(\mathbf{a})$ iff $\mathsf{ch}_{\mathcal{O}}(D) \models q(\mathbf{a})$.

There are several variants of the chase; here we describe the *oblivious chase*. Let $(\mathsf{ch}_k(\mathcal{O}, D))_{k \geq 0}$ be a sequence of structures such that $\mathsf{ch}_0(\mathcal{O}, D) = D$. Then, for every $i > 0$, $\mathsf{ch}_i(\mathcal{O}, D)$ is obtained from $\mathsf{ch}_{i-1}(\mathcal{O}, D)$ by considering all homomorphisms h from the body of some tgd $\phi(\mathbf{x}, \mathbf{y}) \to \exists \mathbf{z} \psi(\mathbf{x}, \mathbf{z})$ in $\mathcal{O}$ to $\mathsf{ch}_{i-1}(\mathcal{O}, D)$ s.t. at least one atom from ϕ is mapped by h into a fact from $\mathsf{ch}_{i-1}(\mathcal{O}, D) \setminus \mathsf{ch}_{i-2}(\mathcal{O}, D)$, and adding to $\mathsf{ch}_i(\mathcal{O}, D)$ all facts obtained from atoms in $\psi(\mathbf{x}, \mathbf{z})$ by replacing each $x \in \mathbf{x}$ with $h(x)$ and each $z \in \mathbf{z}$ with some fresh constant. Then, $\mathsf{ch}_{\mathcal{O}}(D) = \bigcup_{k \geq 0} \mathsf{ch}_k(\mathcal{O}, D)$. Note that $\mathsf{ch}_{\mathcal{O}}(D)$ might be infinite.

A tgd for which all universally quantified variables occur in a body atom is *guarded*. A *GTGD program* is a set of guarded tgds. Unlike evaluation of OMQs based on unrestricted tgds, evaluation of OMQs from (GTGD, UCQ) is decidable [10]. A GTGD program in which all tgds have universally quantified variables only is a *guarded Datalog (GDLog)* program. For every GDLog ontology $\mathcal{O}$ and every database D, $\mathsf{ch}_{\mathcal{O}}(D)$ is finite; furthermore, for every fact $r(\mathbf{a}) \in \mathsf{ch}_{\mathcal{O}}(D)$, there exists a guarded set $\mathbf{a}'$ over D such that $\mathbf{a} \subseteq \mathbf{a}'$.

Parameterized Complexity. For Σ some finite alphabet, a *parameterized problem* is a tuple (P, κ), where $P \subseteq \Sigma^*$ is a problem, and $\kappa : \Sigma^* \to \mathbb{N}$ is a PTIME computable function called the *parameterization* of P. Such a parameterized problem is *fixed-parameter tractable* if there exists an algorithm for deciding P for an input $x \in \Sigma^*$ in time $f(\kappa(x))poly(|x|)$, where f is a computable function and *poly* is a polynomial. The class of all fixed-parameter tractable problems is denoted as FPT.

Given two parameterized problems (P_1, κ_1) and (P_2, κ_2) over alphabets Σ_1 and Σ_2, an *fpt-reduction* from (P_1, κ_1) to (P_2, κ_2) is a function $R : \Sigma_1^* \to \Sigma_2^*$ with the following properties:

1. $x \in P_1$ iff $R(x) \in P_2$, for every $x \in \Sigma_1^*$,
2. there exists a computable function f such that $R(x)$ is computable in time $f(\kappa_1(x))poly(|x|)$,
3. there exists a computable function g such that $\kappa_2(R(x)) \leq g(\kappa_1(x))$, for all $x \in \Sigma_1^*$.

Downey and Fellows [20] defined a hierarchy of parameterized complexity classes $\mathrm{W}[0] \subseteq \mathrm{W}[1] \subseteq \mathrm{W}[2] \ldots$, where $\mathrm{W}[0] = \mathrm{FPT}$ and each inclusion is believed to be strict. Each class W[i], with $i \geq 0$, is closed under fpt-reductions.

A class of interest for us is W[1] as under the assumption that $\mathrm{FPT} \neq \mathrm{W}[1]$, it is possible to establish intractability results (non-membership to FPT) for parameterized problems. A well-known W[1]-complete problem is the parameterized k-clique problem, where the parameter is k: for an input (G, k), with G a graph and $k \in \mathbb{N}^*$, it asks whether G has a k-clique. A stronger assumption than $\mathrm{FPT} \neq \mathrm{W}[1]$, which is widely believed to hold and can be used to establish intractability results, is the *Exponential Time Hypothesis*: it states that 3-SAT with n variables cannot be decided in $2^{o(n)}$ time [26].

Structural Measures: Treewdith, Submodular Width. A *hypergraph* is a pair $H = (V, E)$ with V a set of *nodes* and $E \subseteq 2^V \setminus \{\emptyset\}$ a set of *edges*. A *tree decomposition* of H is a pair $\delta = (T_\delta, \chi)$, with $T_\delta = (V_\delta, E_\delta)$ a tree, and χ a labeling function $V_\delta \to 2^V$ such that:

1. $\bigcup_{t \in V_\delta} \chi(t) = V$.
2. If $e \in E$, then $e \subseteq \chi(t)$ for some $t \in V_\delta$.
3. For each $v \in V$, the set of nodes $\{t \in V_\delta \mid v \in \chi(t)\}$ induces a connected subtree of T_δ.

The *treewidth* of H, $\mathsf{TW}(H)$, is the smallest k such that there exists a tree decomposition (T_δ, χ) of H, with $T_\delta = (V_\delta, E_\delta)$, such that for every $t \in V_\delta$, $|\chi(t)| \leq k$. A function $f : 2^V \to \mathbb{R}_{\geq 0}$ is *submodular* if $f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y)$. It is *edge-dominated* if $f(e) \leq 1$ for all $e \in E$. The *submodular width* of H, $\mathsf{SMW}(H)$, is the smallest k such that for every monotone submodular edge-dominated function f, for which $f(\emptyset) = 0$, there exists a tree decomposition (T_δ, χ) of H, with $T_\delta = (V_\delta, E_\delta)$, such that $f(\chi(t)) \leq k$ for all $t \in V_\delta$.

Every structure I has an associated hypergraph $H_I = (V, E)$, with $V = \mathsf{dom}(I)$ and E the set of guarded sets in I. A tree decomposition $\delta = (T_\delta, \chi)$ of I, with $T = (V_\delta, E_\delta)$, is a tree decomposition of H_I. We say that δ is *guarded* if for every $v \in V_\delta$, $\chi(v)$ is a guarded set in I: $\chi(v) \in E$. Let X range over $\{\mathsf{TW}, \mathsf{SMW}\}$. For a structure I, $\mathsf{X}(I) = \mathsf{X}(H_I)$, for a CQ q, $\mathsf{X}(q) = \mathsf{X}(H_q)$, while for a UCQ q', $\mathsf{X}(q') = \mathsf{max}_{q \text{ is a CQ in } q'}(X(q))$. Finally, for an OMQ $Q = (\mathcal{O}, \mathbf{S}, q)$, with q a UCQ, $\mathsf{X}(Q) = \mathsf{X}(q)$.

3 Normalizing OMQs: Characteristic Databases and Covers

As seen earlier, *equivalence-based measures* frequently play a role in fpt characterizations. Following [6, 4, 3], we refer to such measures as *semantic measures.* In particular, for an OMQ $Q \in (\mathcal{L}, \mathrm{UCQ})$, with $\mathcal{L}$ an ontology language, and a structural measure $\mathsf{X} \in \{\mathsf{TW}, \mathsf{SMW}\}$, the semantic X-width of Q is the smallest k such that there exists an OMQ Q' from $(\mathcal{L}, \mathrm{UCQ})$ with $Q \equiv Q'$ and $\mathsf{X}(Q') = k$. A class $\mathbb{Q}$ of OMQs has *bounded semantic* X*-width* if there exists $k > 0$ such that every OMQ in $\mathbb{Q}$ has semantic X-width at most k. However, in order to establish such characterizations, an important issue is finding witnesses of (bounded) semantic measures, i.e. classes of OMQs of low measures which are equivalent to the original ones.

For CQs, cores serve as witnesses for semantic treewidth [24] and also for semantic submodular width [16]. Thus, as concerns classes of CQs of bounded semantic X-width, with $\mathsf{X} \in \{\mathsf{TW}, \mathsf{SMW}\}$, the class of cores of CQs from the original class serves as a witness, i.e. it has actual bounded X-width. As concerns OMQs based on UCQs, resorting to cores of CQs in UCQs does not necessarily lead to witnesses of low width. The ontology also plays a role in lowering semantic measures. For examples of this phenomenon as concerns semantic treewidth, see [4, 3]. Here, we show how the ontology influences semantic submodular width:

▶ **Example 1.** For R a binary relational symbol, $i \in \mathbb{N}$, with $i > 1$, and $\mathbf{x}_i$ an i-tuple of variables, we denote with $\psi_i^R(\mathbf{x}_i)$ the formula $R(x_1, x_2) \wedge R(x_1, x_3) \wedge \cdots \wedge R(x_{i-1}, x_i)$, i.e. the hypergraph associated to ψ_i^R is the i-clique. Let T be a binary relational symbol and $\mathbb{Q}$ be the class of OMQs $(Q_i)_{i>1}$, with $Q_i = (\mathcal{O}_i, \mathbf{S}_i, q_i)$, where:

$$\mathcal{O}_i = \{S_i(\mathbf{x}_i) \to \psi_i^R(\mathbf{x}_i)\} \qquad \mathbf{S}_i = \{S_i, T\} \qquad q_i = \exists \mathbf{x}_i\ \psi_i^R(\mathbf{x}_i) \wedge \psi_i^T(\mathbf{x}_i)$$

Then $\mathbb{Q}$ has unbounded submodular width. As for every $i > 1$, $H_{D[q_i]}$ is the i-clique, every tree decomposition (T, χ) of $H_{D[q_i]}$, with $T = (V, E)$, must contain some node $t \in V$ such that $\chi(t) = \mathbf{x}_i$. Let $f : 2^{\mathbf{x}_i} \to \mathbb{R}_{\geq 0}$ be the monotone submodular function $f(X) = |X|/2$. Then f is also edge-dominated with respect to $D[q_i]$, and its minimum over all tree decompositions of $H_{D[q_i]}$ is $i/2$. Thus, $\mathsf{SMW}(Q_i) \geq i/2$, for every $i > 1$.

On the other hand, for every $i > 1$, R is not part of the schema $\mathbf{S}_i$ and the only way to derive it is using the unique tgd from $\mathcal{O}_i$. Thus, every $\mathbf{S}_i$-database D_i such that $D_i \models Q_i$ must contain an atom of the form $S_i(\mathbf{c}_i)$, where $\mathbf{c}_i$ is an i-tuple of constants. Then, for every $i > 0$, Q_i is equivalent to the OMQ $Q_i' = (\mathcal{O}_i, \mathbf{S}_i, q_i')$, with $q_i' = \exists \mathbf{x}_i\ S_i(\mathbf{x}_i) \wedge \psi_i^R(\mathbf{x}_i) \wedge \psi_i^T(\mathbf{x}_i)$.

Let $\mathbb{Q}'$ be the class of OMQs $(Q'_i)_{i>1}$. As for every $i > 1$, q'_i is guarded, i.e. it contains an atom $S_i(\mathbf{x}_i)$ which has as terms $\mathsf{var}(q'_i)$, it follows that $\mathsf{SMW}(q'_i) \leq 1$: this is due to the fact that only edge-dominated functions are considered when defining the submodular width and thus the guard ensures that for every such function f, $f(\mathbf{x}_i) \leq 1$. Thus, $\mathbb{Q}'$ has bounded submodular width and $\mathbb{Q}$ has bounded semantic submodular width.

Example 1 shows that submodular width can be lowered by adding extra atoms to CQs in the original OMQ. In this section, we normalize OMQs from (GDLog, UCQ) by extending (images of) CQs in the original OMQ with facts occurring in databases which entail the OMQ. We obtain equivalent OMQs called *covers*. The purpose of the added atoms is to provide guards for atoms in CQs and to potentially lower submodular width, as in Example 1. Intuitively, by adding such guards, the submodular width is decreased as the space of edge-dominated submodular functions is shrunk.

At the same time, when constructing covers, we do not want to add too many atoms: adding cliques of unbounded size (without a guard) would obviously not decrease submodular width. As such, we use as the basis for the construction only a set of selected databases which are small w.r.t. the homomorphism order and which we call *extended characteristic databases*. Ideally, we would consider only databases which entail the OMQ and which are minimal w.r.t. the homomorphism order. However, for a given OMQ there might be no such minimal databases:

▶ **Example 2.** Let $Q = (\mathcal{O}, \mathbf{S}, q)$ be the following OMQ from (GDLog, UCQ):

$$\begin{aligned}
\mathcal{O} &= \{A(x) \wedge R(x,y) \rightarrow A(y),\ A(x) \rightarrow B(x)\} \\
\mathbf{S} &= \{A, R, C\} \\
q &= \exists x\ B(x) \wedge R(x,x) \wedge C(x)
\end{aligned}$$

Also, for every $n \in \mathbf{N}$, let D_n be the $\mathbf{S}$-database $\{A(x_0), C(x_n), R(x_n, x_n)\} \cup \{R(x_i, x_{i+1}) \mid 0 \leq i < n\}$. Then, for every $n \in \mathbf{N}$, $D_n \models Q$, D_n is a core, $D_{n+1} \rightarrow D_n$, and $D_n \not\rightarrow D_{n+1}$. Thus, D_n is not minimal w.r.t. $\rightarrow$. Furthermore, for every $\mathbf{S}$-database D, it can be checked that $D \models Q$ implies that there exists $n \in \mathbf{N}$ such that $D_n \rightarrow D$, and thus D is not minimal w.r.t. the homomorphism order.

It is also not possible to use the set of canonical databases of CQs occurring in the OMQ as a basis for the construction, as in the presence of schema restrictions, those databases might not be over the expected schema. As such, to identify a set of selected databases, we apply successive transformations on the set of databases which entail an OMQ. The transformations are described in Sections 3.1, 3.2, and 3.3. Section 3.4 brings all the concepts together to define (extended) characteristic databases and covers.

3.1 Query Initial Databases

In our quest to refine databases which entail an OMQ we look at how CQs from the OMQ map into the chase of a given database. In the following, a *contraction* of a CQ q is a CQ obtained from q by variable identification.

▶ **Definition 3.** *For $Q = (\mathcal{O}, \mathbf{S}, q)$ from (GDLog, UCQ) and D an $\mathbf{S}$-database such that $D \models Q$, we say that D is* query-initial (qi) *w.r.t. Q if for every $\mathbf{S}$-database D' such that $D' \rightarrow D$ and $D' \models Q$, and every contraction p of some CQ in q it is the case that: $p \rightarrow \mathsf{ch}_{\mathcal{O}}(D')$ iff $p \rightarrow \mathsf{ch}_{\mathcal{O}}(D)$.*

Thus, qi databases entail an OMQ and are minimal w.r.t. the set of contractions which map into their chase. They are related to injectively only databases which have been introduced in [4]. By a simple induction argument, it can be shown that:

▶ **Lemma 4.** *Let Q be an $\mathbf{S}$-OMQ from (GDLog, UCQ), and D an $\mathbf{S}$-database with $D \models Q$. If D is qi w.r.t. Q, then for every database $D' \to D$, $D' \models Q$ implies D' is qi w.r.t. Q. If D is not qi w.r.t. Q, there exists a database $D' \to D$ such that $D' \models Q$ and D' is qi w.r.t. Q.*

▶ **Example 5.** Let $Q = (\mathcal{O}, \mathbf{S}, q)$ be the OMQ with:

$$\mathcal{O} = \{U(x,y,z) \wedge V(x,z) \to T(x,z), W(x,y,z) \to S(y,z)\}$$
$$\mathbf{S} = \{R, U, V, W\}$$
$$q = \exists x, y, z\ R(x,y) \wedge S(y,z) \wedge T(z,x)$$

Also, let D_1 and D_2 be the $\mathbf{S}$-databases: $D_1 = \{R(a,b), W(d,b,a), U(a,d,a), V(a,a)\}$ and $D_2 = \{R(a,b), W(d,b,c), U(c,d,a), V(c,a)\}$. Then $D_2 \to D_1$.

Let q' be the following contraction of q: $q' = \exists x, y\ R(x,y) \wedge S(y,x) \wedge T(x,x)$ (obtained by identification of x and z in q). Then both q and q' map into $\mathsf{ch}_{\mathcal{O}}(D_1)$, but only q maps into $\mathsf{ch}_{\mathcal{O}}(D_2)$. Thus, D_1 is not minimal w.r.t the set of contractions which map into its chase, and consequently it is not qi w.r.t. Q. However, D_2 is qi w.r.t. Q.

3.2 Guarded Unravelings

Another concept which will turn useful for defining characteristic databases is that of *guarded unraveling*. The operation was first introduced in [22]. Given a database D and a guarded set set $\mathbf{a}$ over D, it unfolds D into a (potentially infinite) structure $I^{\mathbf{a}}$, the *guarded unraveling of D at* $\mathbf{a}$, which admits a guarded tree decomposition $\delta = (T, \chi)$, with $T = (V, E)$. The vertices $t \in V$ of T are sequences of the form $\mathbf{a}_0 \dots \mathbf{a}_n$, where $\mathbf{a}_0 = \mathbf{a}$, and $\mathbf{a}_1, \dots, \mathbf{a}_n$ are maximal guarded sets in D such that $\mathbf{a}_i \cap \mathbf{a}_{i+1} \neq \emptyset$, and $\mathbf{a}_i \neq \mathbf{a}_{i+1}$, for every $0 \leq i < n$. For every $t_1, t_2 \in V$, $(t_1, t_2) \in E$ iff $t_2 = t_1\mathbf{b}$, for some maximal guarded set $\mathbf{b}$ in D.

Intuitively, we unravel D guided by δ: each vertex $t \in V$ of the form $\mathbf{a}_0 \dots \mathbf{a}_n$ can be seen as a local representative (in the unraveling) of the the maximal guarded set $\mathbf{a}_n$ in D; as such, $\chi(t)$ contains copies of the constants from $\mathbf{a}_n$. Assuming we build T in an inductive fashion, we set $\chi(r) = \mathbf{a}$, where r is the root of T and then, for each newly created $t \in V$ of the form $\mathbf{a}_1, \dots, \mathbf{a}_n$, we introduce fresh copies of constants a which occur in the guarded set $\mathbf{a}_n$ corresponding to t, but do not occur in the guarded set $\mathbf{a}_{n-1}$ corresponding to its predecessor node $t' = \mathbf{a}_1 \dots \mathbf{a}_{n-1}$. We denote such a fresh copy of a, introduced while creating t, as a_t. Then, we set $\chi(t) = \{a_{t'} \mid a \in \mathbf{a}_n \cap \mathbf{a}_{n-1}\} \cup \{a_t \mid a \in \mathbf{a}_n \setminus \mathbf{a}_{n-1}\}$. We also define D_t to be a copy of the database $I|_{\mathbf{a}_n}$, where every $a \in \mathbf{a}_n$ has been replaced with the corresponding a_t or $a_{t'}$ from $\chi(t)$. $I^{\mathbf{a}}$ is the union of all databases D_t with $t \in V$.

Guarded unravelings have the property that for every OMQ Q' from (GDLog, AQ), every database D, every guarded set $\mathbf{a}$ in D, and every tuple $\mathbf{a}' \subseteq \mathbf{a}$: $D \models Q'(\mathbf{a}')$ implies $I^{\mathbf{a}} \models Q'(\mathbf{a}')$ [22]. It can also easily be seen that, due to the existence of a guarded tree decomposition, the submodular width of $I^{\mathbf{a}}$ is 1. As such, for a given OMQ $Q = (\mathcal{O}, \mathbf{S}, q)$ from (GDLog, UCQ) and some $\mathbf{S}$-database D such that $D \models Q$, we will use them to disentangle parts of D which are needed to entail specific atoms from some CQ in Q, i.e. we will replace some parts of D with corresponding guarded unravelings.

In general, for a guarded set $\mathbf{a}$ in D, $I^{\mathbf{a}}$ is infinite and thus cannot be used straightaway. By compactness, there exists a finite subset (a database) $D^{\mathbf{a}}$ which fulfills the same property w.r.t. entailment of OMQs based on AQs. We fix such a database $D^{\mathbf{a}}$. However, as we want to unravel databases which entail OMQs as much as possible, we place a stronger requirement on the chosen $D^{\mathbf{a}}$: for every Boolean sub-query p' of some CQ p in q, whenever $I^{\mathbf{a}} \models (\mathcal{O}, \mathbf{S}, p')$, it must be the case that $D^{\mathbf{a}} \models (\mathcal{O}, \mathbf{S}, p')$. As this requirement is relative to Q, we will refer to $D^{\mathbf{a}}$ as the *guarded unraveling of* D *at* $\mathbf{a}$ *w.r.t.* Q. However, Q will be clear in most cases from the context so it will be omitted.

3.3 Diversifications

A *diversification* of a database is essentially a database which maps into the original one in a specific way. For a function f and S a subset of its domain, we denote with $f|_S$ the restriction of f on S. Also, we denote with $\mathsf{ran}(f)$ the range of f. A homomorphism h from a structure A to a structure B is said to be *injective on guarded sets (i.g.s.)* if $h|_{\mathbf{a}}$ is injective, for every guarded set $\mathbf{a}$ in A. For a database D, a constant $c \in \mathsf{dom}(D)$ is *isolated in* D if it occurs in a single fact in D. The *kernel* of a database D, $\mathsf{ker}(D)$, is the set of non-isolated constants in D.

▶ **Definition 6.** *A* diversification *of a database D_0 is a tuple $(D, \uparrow)$, where D is a database which maps into D_0 via the homomorphism $\uparrow$ such that:*
1. $\uparrow_{|\mathsf{ker}(D)}$ *is injective, and*
2. $\uparrow$ *is i.g.s.*

We write $D \preceq D_0$, whenever there exists a diversification $(D, \uparrow)$ of D_0.

▶ **Example 7.** Let $D_2 = \{R(a,b), W(d,b,c), U(c,d,a), V(c,a)\}$ be the database from Example 5. Also, let D be the $\mathbf{S}$-database: $\{R(a,b), W(d_1,b,c), U(c,d_2,a)\}$. Then, the mapping $\uparrow\colon \mathsf{dom}(D) \to \mathsf{dom}(D_2)$, which is the identity on $\{a,b,c\}$ and maps d_1 and d_2 to d is a homomorphism from D to D_2. As $\mathsf{ker}(D) = \{a,b,c\}$ and $\uparrow$ is i.g.s., it follows that $(D, \uparrow)$ is a diversification of D_2 and thus $D \preceq D_2$.

For a database D which entails an OMQ Q, we will use diversifications of D in conjunction with guarded unravelings of D w.r.t. Q to construct new databases which map into D and which still entail Q. We start with a construction which glues guarded unravelings of some database D_0 w.r.t. Q to a database D which maps into D_0 via some homomorphism $\uparrow$ which is i.g.s. (in this case $(D, \uparrow)$ need not be a diversification of D_0). We denote with $\mathsf{ext}_Q(D, \uparrow, D_0)$ the database obtained from D by adding for each maximal guarded set $\mathbf{a}$ in D the database $D_0^{\mathbf{a}}$ obtained from the guarded unraveling $D_0^{\uparrow(\mathbf{a})}$ of D_0 w.r.t. Q by renaming the constants in $\uparrow(\mathbf{a})$ to those in $\mathbf{a}$. When Q is clear from the context, we write $\mathsf{ext}(D, \uparrow, D_0)$.

▶ **Definition 8.** *For Q an $\mathbf{S}$-OMQ from (GDLog, UCQ) and D_0 an $\mathbf{S}$-database with $D_0 \models Q$:*
1. $\mathsf{div}(D_0, Q)$ *is the set of diversifications $(D, \uparrow)$ of D_0 for which* $\mathsf{ext}(D, \uparrow, D_0) \models Q$;
2. *a diversification $(D, \uparrow)$ from $\mathsf{div}(D_0, Q)$ is* minimal w.r.t. Q *if D is a core and there is no other diversification $(D', \downarrow)$ from $\mathsf{div}(D_0, Q)$ such that $D' \preceq D$, and $D \not\preceq D'$;*
3. $\mathsf{mdiv}(D_0, Q)$ *is the set of all minimal diversifications of D_0 w.r.t. Q.*

Intuitively, for a minimal diversification $(D, \uparrow)$ of D_0 w.r.t. Q and the ensuing database $\mathsf{ext}(D, \uparrow, D_0)$, the D-part of $\mathsf{ext}(D, \uparrow, D_0)$ is important for preserving some part of the hypergraph of D_0 needed to satisfy the geometry (hypergraph) of a query. Guarded unravelings provide the necessary information to entail specific atoms in the query.

▶ **Example 9.** Let $Q = (\mathcal{O}, \mathbf{S}, q)$ and D_2 be as in Example 5. Also let $(D, \uparrow)$ be the diversification of D_2 introduced in Example 7 and let $D^+ = \mathsf{ext}(D, \uparrow, D_2)$ be the database obtained from D by adding guarded unravelings of D_2. We do not explicitly construct the guarded unravelings, but note that $D_2^{(a,c,d)}$ will contain the fact $V(c,a)$. Let $D' = D \cup \{V(c,a)\}$. Then $D' \subseteq D^+$ and $D' \models Q$, thus $D^+ \models Q$. Thus, $(D, \uparrow) \in \mathsf{div}(D_2, Q)$.

In fact, $(D, \uparrow) \in \mathsf{mdiv}(D_2, Q)$. To see why that is the case, we observe that every database D_3 for which $D_3 \preceq D$, but $D \not\preceq D_3$, is isomorphic to some database obtained from D by dropping facts or renaming non-isolated constants – renaming isolated constants would simply result into an equivalent database. It can be checked that for any such database D_3, the addition of guarded unravelings of D_2 is no longer enough to entail Q. Consider for example the database $D_3 = \{R(a, b_1), W(d_1, b_2, c), U(c, d_2, a)\}$ obtained from D by renaming the two occcurrences of b as b_1 and b_2. Also, let $\downarrow$ be the homomorphism from D_3 to D_2 which is the identity on a and c and maps b_1 and b_2 to b and d_1 and d_2 to d. Then, $\mathsf{ext}(D_3, \downarrow, D_2) \not\models Q$: while $\mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D_3, \downarrow, D_2))$ will contain atoms over R, S, and T (as requested by q), the underlying hypergraph structure of D needed to entail q is lost.

On the other hand, q maps into $\mathsf{ch}_{\mathcal{O}}(D^+)$ via a homomorphism h which maps x, y, z to a, b, c. The hypergraph H_q is isomorphic to a sub-hypergraph of H_D. In this sense, H_D preserves the structure of D_2 needed to entail q. At the same time, the atoms added by guarded unravelings, like $V(c, a)$, allow the entailment of particular atoms from q, like $T(x, z)$.

We next show how given a homomorphism h from a CQ p in an OMQ Q to the chase of a database of the form $\mathsf{ext}(D, \uparrow, D_0)$, it is possible to construct a diversification of D, $(D', \downarrow)$, such that the database obtained by extending D' with guarded unravelings of D_0 according to the composition homomorphism $\uparrow \circ \downarrow$ from D' still entails the OMQ Q.

► **Lemma 10.** *Let $Q = (\mathcal{O}, \mathbf{S}, q)$ be an OMQ from (GDLog, UCQ), D and D_0 be $\mathbf{S}$-databases, and $\uparrow$ a homomorphism from D to D_0 which is i.g.s. such that $\mathsf{ext}(D, \uparrow, D_0) \models Q$. Also, let h be a homomorphism from some CQ p in q to $\mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D, \uparrow, D_0))$. Then, there exists a diversification $(D', \downarrow)$ of D such that:*
1. *$\downarrow$ is the identity function on $\mathsf{ker}(D')$;*
2. *$\mathsf{ker}(D') \subseteq \mathsf{ran}(h) \cap \mathsf{dom}(D)$;*
3. *$\mathsf{ext}(D', \uparrow \circ \downarrow, D_0) \models Q$.*

Proof. We start by partitioning the set of variables from the CQ p, $\mathsf{dom}(p)$, into maximal sets $A_0, A_1, \dots A_n$, such that:
1. $h(A_0) \subseteq \mathsf{dom}(D)$;
2. for every $i > 0$, $h(A_i) \subseteq \mathsf{dom}(D_0^{\mathbf{a}_i}) \setminus \mathsf{dom}(D)$, where $\mathbf{a}_i$ is some maximal guarded set in D and $D^{\mathbf{a}_i}$ is the corresponding guarded unraveling of D_0 which has been added to D during the construction of $\mathsf{ext}(D, \uparrow, D_0)$.

Thus, A_0 is the set of all variables which map into $\mathsf{dom}(D)$, and each A_i is a maximal set of variables which map into constants from some guarded unraveling of D_0 which are not from D. Note that by maximality of the sets A_i, it follows that $\mathbf{a}_i \neq \mathbf{a}_j$, whenever $i \neq j$.

We now proceed to defining D'. We start by initializing it as the set of atoms $\{r(\mathbf{c}) \in D \mid \mathbf{c} \subseteq \mathsf{ran}(h)\}$. These are exactly the atoms for which all their terms are from $h(A_0)$. Then, for every i, with $1 \leq i \leq n$, there must be some atom $r(\mathbf{a}_i)$ in D (where $\mathbf{a}_i$ is as above in the definition of the sets A_i). Note that there might be more than one such atom, however it is enough to select one. We rename all constants from $\mathbf{a}_i$ which are not from $\mathsf{ran}(h)$ as fresh constants and add the atom to D'. Intuitively, each such atom will serve as a skeleton, as a guard for adding guarded unravelings of D_0 to D'.

By construction, $\mathsf{dom}(D) \cap \mathsf{ran}(h) \subseteq \mathsf{dom}(D')$. All other constants from $\mathsf{dom}(D')$ are fresh, and thus isolated. Thus, $\mathsf{ker}(D') \subseteq \mathsf{dom}(D) \cap \mathsf{ran}(h)$ (Point (2) of the Lemma). We define $\downarrow$ as a mapping from $\mathsf{dom}(D')$ to $\mathsf{dom}(D)$ which is the identity on $\mathsf{ran}(h) \cap \mathsf{dom}(D)$ and which maps fresh constants to original constants for the remaining elements of $\mathsf{dom}(D')$. It can be verified easily that $(D', \downarrow)$ is a diversification of D and that $\downarrow$ is the identity on $\mathsf{ker}(D')$ – Point (1) of the Lemma. It remains to show Point (3).

Let $g = \uparrow \circ \downarrow$ and $D^+ = \mathsf{ext}(D', g, D_0)$. For every i with $0 < i \le n$, we define an isomorphism κ_i from the database $D_0^{\mathbf{a}_i}$, the guarded unraveling of D_0 added to the maximal guarded set $\mathbf{a}_i$ during the construction of $\mathsf{ext}(D, \uparrow, D_0)$ to the database $D_0^{\mathbf{b}_i}$, the copy of $D_0^{\mathbf{a}_i}$ added to D' during the construction of D^+, in which constants from $\mathbf{a}_i$ have been replaced with constants from $\mathbf{b}_i$. Note that $\kappa_i(\mathbf{a}_i) = \mathbf{b}_i$ and $\downarrow(\mathbf{b}_i) = \mathbf{a}_i$.

We construct a mapping h' from $\mathsf{dom}(p)$ to $\mathsf{dom}(D^+)$ as follows:

$$h'(x) = \begin{cases} h(x) & \text{if } x \in A_0 \\ \kappa_i(h(x)) & \text{if } x \in A_i \end{cases}$$

Due to the property of guarded unravelings to preserve atomic consequences, it can be shown that for every i with $0 < i \le n$, κ_i is an isomorphism also from the restriction of $\mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D, \uparrow, D_0))$ to $\mathsf{dom}(D_0^{\mathbf{a}_i})$ to the restriction of $\mathsf{ch}_{\mathcal{O}}(D^+)$ to $\mathsf{dom}(D_0^{\mathbf{b}_i})$. Furthermore, again due to the property of guarded unravelings to preserve atomic consequences, it can be shown that for every atom $r(\mathbf{c})$ in $\mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D, \uparrow, D_0))$ for which $\mathbf{c} \subseteq \mathsf{ran}(h)$, it is the case that $r(\mathbf{c})$ in $\mathsf{ch}_{\mathcal{O}}(D^+)$ as well. Then it can be shown that h' is a homomorphism from p to $\mathsf{ch}_{\mathcal{O}}(D^+)$. ◀

Lemma 10 will be useful later as it offers a way to construct databases based on diversifications which are progressively smaller w.r.t. the homomorphism order.

3.4 Characteristic Databases and Covers

In this section we put together the notions introduced in previous subsections to define *(extended) characteristic databases* and *covers* of an OMQ.

▶ **Definition 11.** *For Q an OMQ from (GDLog, UCQ),* the set of characteristic databases for *Q is $\mathbb{D}_Q = \{D \mid (D, \uparrow) \in \mathsf{mdiv}(D_0, Q), D_0$ is qi w.r.t. $Q\}$.* The set of extended characteristic databases for *Q is $\mathbb{D}_Q^+ = \{\mathsf{ext}(D, \uparrow, D_0) \mid (D, \uparrow) \in \mathsf{mdiv}(D_0, Q), D_0$ is qi w.r.t. $Q\}$.*

Thus, a characteristic database D is part of a minimal diversification $(D, \uparrow)$ w.r.t. Q of a database D_0 which is qi w.r.t. Q. It is easy to see that:

▶ **Lemma 12.** *For Q an $\mathbf{S}$-OMQ from (GDLog, UCQ) and D an $\mathbf{S}$-database such that $D \models Q$, there exists a database $D' \in \mathbb{D}_Q^+$ such that $D' \to D$.*

Furthermore, characteristic databases have the following properties:

▶ **Lemma 13.** *Let $Q = (\mathcal{O}, \mathbf{S}, q)$ be an OMQ from (GDLog, UCQ), $\mathsf{ext}(D, \uparrow, D_0)$ be a database from $\mathbb{D}_Q^+$, and h be a homomorphism from a CQ p in q to $\mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D, \uparrow, D_0))$. Then:*
1. *$\mathsf{ker}(D) \subseteq \mathsf{ran}(h)$;*
2. *there exists a computable function f such that $|D| \le f(|Q|)$.*

Proof. We start by showing Point (1). From Lemma 10 we know that there exists a database D' and a homomorphism $\downarrow$ from D' to D such that $(D', \downarrow)$ is a diversification of D. Thus, $D' \preceq D$. As $(D, \uparrow)$ is a diversification of D_0, it follows that $(D', \uparrow \circ \downarrow)$ is a diversification of D_0. From Point (3) of Lemma 10 we obtain that $(D', \uparrow \circ \downarrow)$ is a diversification of D_0 w.r.t. Q. In other words, $D' \in \mathsf{div}(D_0, Q)$. As $D \in \mathsf{mdiv}(D_0, Q)$, and $D' \preceq D$, it must be the case that $D \preceq D'$ (otherwise, D would not be minimal). Thus, $D \leftrightarrow D'$.

From Point (1) of Lemma 10, we know that D' maps into D via the function $\downarrow$ which is the identity on $\mathsf{ker}(D')$. Thus, $\mathsf{ker}(D') \subseteq \mathsf{ker}(D)$ (a non-isolated constant, i.e. a constant which occurs in $\mathsf{ker}(D')$ can only map into another non-isolated constant, thus a constant from $\mathsf{ker}(D)$). As D is a core, it is the case that D maps into D' only via injective homomorphisms: otherwise, the composition of a non-injective homomorphism from D to D' with the homomorphism $\downarrow$ from D' to D would lead to a non-injective endomorphism on D which is in contradiction to the fact that D is a core. As every homomorphism from D to D' maps $\mathsf{ker}(D)$ into $\mathsf{ker}(D')$ injectively, it is the case that $|\mathsf{ker}(D)| \leq |\mathsf{ker}(D')|$. Thus, $\mathsf{ker}(D') = \mathsf{ker}(D)$. Finally, from Point (2) of Lemma 10 we know that $\mathsf{ker}(\mathsf{D}') \subseteq \mathsf{ran}(h) \cap \mathsf{dom}(D)$, and thus $\mathsf{ker}(D) \subseteq \mathsf{ran}(h)$.

Point (2) follows from Point (1) and from the fact that D is a core: as $|\mathsf{ker}(D)|$ is bounded in $|Q|$, $|D|$ will be bounded in $|Q|$ as well. ◀

We next define the *cover* $Q_c = (\mathcal{O}, \mathbf{S}, q_c)$ of an OMQ $Q = (\mathcal{O}, \mathbf{S}, q)$ by using the set of extended characteristic databases for Q, $\mathbb{D}^+_Q$. To construct CQs in q_c, we will consider images of CQs from q in the chase of some database D^+ from $\mathbb{D}^+_Q$ together with extra atoms (facts)[1] from D^+ which will guard atoms in the CQ image. We want to include guards from D^+ which cover as many atoms as possible from the image of the CQ, and thus to potentially decrease submodular width as in Example 1. Formally:

▶ **Definition 14.** *For $Q = (\mathcal{O}, \mathbf{S}, q)$ an OMQ from (GDLog, UCQ), its* cover *is the OMQ $Q_c = (\mathcal{O}, \mathbf{S}, q_c)$, with q_c the union of all CQs p_c with $D[p_c]$ of the form $h(p) \cup S$, where:*

1. *h is a homomorphism from some CQ p in q_c to $\mathsf{ch}_{\mathcal{O}}(D^+)$, where D^+ is a database of the form $\mathsf{ext}(D, \uparrow, D_0)$ from $\mathbb{D}^+_Q$;*
2. *S is a minimal set of atoms such that:*
   - **a.** *$D \subseteq S \subseteq D^+$;*
   - **b.** *for every atom $r(\mathbf{a})$ from $h(p)$, there exists an atom $r'(\mathbf{a}')$ from S such that $\mathbf{a} \subseteq \mathbf{a}'$ and $\mathbf{a}'$ is a maximal guarded set in D^+.*

The set of atoms S in Definition 14 is the set of guards added to the image $h(p)$ of a CQ p in q. By including D, S trivially guards every maximal set of atoms S' from $h(p)$ for which $\mathsf{dom}(S')$ is a guarded set in D. All other atoms from $h(p)$ will map into the guarded unraveling part of (the chase of) D^+. For every maximal set S' of such atoms for which $\mathsf{dom}(S')$ is a guarded set in D^+, there exists a unique atom $r'(\mathbf{a}')$ in D^+ such that $\mathsf{dom}(S') \subseteq \mathbf{a}'$. Thus, the definition insures that “maximal” guards, which cover as many atoms as possible, are added to $h(p)$.

▶ **Example 15.** Let $\mathbb{Q}$ be as in Example 1. For every $i > 1$, let $D_{i,0}$ be the $\mathbf{S}_i$-database: $\{S_i(\mathbf{a}_i), T(a_1, a_2), T(a_1, a_3), \ldots T(a_{i-1}, a_i)\}$. We have that $D_{i,0} \models Q_i$ and furthermore, $D_{i,0}$ is qi w.r.t. Q_i: q_i is the only contraction mapping into $\mathsf{ch}_{\mathcal{O}_i}(D_{i,0})$. Let $D_i = \{S(\mathbf{a}_i)\}$. Then $(D_i, \uparrow) \in \mathsf{mdiv}(D_{i,0}, Q)$, where $\uparrow$ is the identity function on $\mathbf{a}_i$: any guarded unraveling of $D_{i,0}$ at $\mathbf{a}_i$ will add back all the facts from $D_{i,0}$. Thus, $D_i \in \mathbb{D}_{Q_i}$ and for every database of the form $\mathsf{ext}(D_i, \uparrow, D_{i,0})$, $D_{i,0} \subseteq \mathsf{ext}(D_i, \uparrow, D_{i,0})$ (there might be more facts in $\mathsf{ext}(D_i, \uparrow, D_{i,0})$). In the following we fix such a database $\mathsf{ext}(D_i, \uparrow, D_{i,0})$ and refer to it as D^+_i.

The CQ q_i maps then into $\mathsf{ch}_{\mathcal{O}_i}(D^+_i)$ via a homomorphism h with range $\mathbf{a}_i$. We construct a CQ p_c belonging to the cover $Q_{i,c}$ of Q_i based on h and D^+_i: $D[p_c]$ will be the union of $\psi^R_i(\mathbf{a}_i) \wedge \psi^T_i(\mathbf{a}_i)$, the image of q_i under h, with the singleton set $\{S(\mathbf{a}_i)\}$ from D^+_i: $D[p_c] = \psi^R_i(\mathbf{a}_i) \wedge \psi^T_i(\mathbf{a}_i) \wedge S(\mathbf{a}_i)$. We note that $S(\mathbf{a}_i)$ is a guard for all atoms from $h(q_i)$ and $D[p_c]$ is isomorphic to $D[q'_i]$ from Example 1. It can be shown that p_c is the unique such CQ (up to isomorphism) and thus $Q_{i,c}$ is identical to the OMQ Q'_i from Example 1.

---

[1] As we construct a CQ by conjoining the image of an original CQ into a database with more facts from the database, the distinction between facts and atoms is blurred.

As the next lemma shows, covers are equivalent to original OMQs. Intuitively, this is the case as we only add atoms to CQs in the original OMQ from databases which entail the OMQ and which are small w.r.t. the homomorphism order:

▶ **Lemma 16.** *For $Q = (\mathcal{O}, \mathbf{S}, q)$ an OMQ from (GDLog, UCQ), and Q_c its cover, $Q \equiv Q_c$.*

Proof. It is clear that $Q_c \subseteq Q$. We show that $Q \subseteq Q_c$. For D an $\mathbf{S}$-database such that $D \models Q$, there exists a database $D_0 \to D$ which is qi w.r.t. Q. Furthermore, there exists a diversification $(D_1, \uparrow) \in \mathsf{mdiv}(D_0, Q)$. Thus, $\mathsf{ext}(D_1, \uparrow, D_0) \in \mathbb{D}^+_Q$, and there exists some CQ p in q s.t. p maps into $\mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D_1, \uparrow, D_0))$ via some homomorphism h. We construct a CQ p_c from q_c as in Definition 14 based on p, h, and $\mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D_1, \uparrow, D_0))$. Then, $D[p_c] \subseteq \mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D_1, \uparrow, D_0))$, thus $p_c \to \mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D_1, \uparrow, D_0))$, and as p_c is a CQ in q_c, it follows that $\mathsf{ext}(D_1, \uparrow, D_0) \models Q_c$. As $\mathsf{ext}(D_1, \uparrow, D_0) \to D_0 \to D$, it follows that $D \models Q_c$. ◀

Thus covers are a good candidate for witnesses of low semantic SMW for OMQs from (GDLog, UCQ). We defer to the next section the result detailing in which way they serve as such witnesses. For now we show that they have finite bounded size and are computable:

▶ **Lemma 17.** *For $Q = (\mathcal{O}, \mathbf{S}, q)$ an OMQ from (GDLog, UCQ) and $Q_c = (\mathcal{O}, \mathbf{S}, q_c)$ its cover, there exists a computable function g such that for every CQ p_c in q_c, $|p_c| \leq g(|Q|)$.*

Proof. Each p_c has the form $h(p) \cup S$, where p is some CQ in q, h a homomorphism from p to $\mathsf{ch}_{\mathcal{O}}(\mathsf{ext}(D, \uparrow, D_0))$, for some $\mathsf{ext}(D, \uparrow, D_0) \in \mathbb{D}^+(Q)$ and S the extension of D with atoms from $\mathsf{ext}(D, \uparrow, D_0) \in \mathbb{D}^+(Q)$ which guard atoms in $h(p)$. As $|h(p)|$ is bounded in $|Q|$, $|D| \leq f(|Q|)$, for some computable function f (from Lemma 13) and S contains at most one atom for each atom in $h(p)$, it follows that $|p_c|$ is bounded in $|Q|$. ◀

▶ **Lemma 18.** *For every OMQ $Q = (\mathcal{O}, \mathbf{S}, q) \in$ (GDLog, UCQ), $Q_c = (\mathcal{O}, \mathbf{S}, q_c)$ is computable.*

Proof Sketch. Let $\mathbf{S}'$ be the extension of $\mathbf{S}$ with relational symbols which occur in Q. Also, let *the diameter of* Q be the maximum between the arity of some relational symbol from $\mathbf{S}$ and the number of variables in some CQ in q. We devise an encoding into Guarded Second Order Logic (GSO) [23] to check whether some structure over the extended signature $\mathbf{S}'$ of size bounded by $g(|Q|)$ (with g as in Lemma 17) is of the right form to be part of q_c. While GSO is in general undecidable, on sparse structures it has the same expressivity as MSO [9] and thus it is decidable. For our purposes, it is possible to restrict to structures of treewidth bounded by the diameter of the given OMQ, which are sparse [18]. ◀

4 Main Results

In this section we establish the main results of our paper: a characterization for fixed-parameter tractability of OMQs from (GDLog, UCQ), a reduction from parameterized uniform CSPs to parameterized OMQs from (GDLog, UCQ), and a semantic characterization for fixed-parameter tractability of OMQs from (GTGD, UCQ). We conclude the section with a discussion concerning covers of OMQs from (GDLog, UCQ): we show that they are indeed witnesses for bounded semantic submodular width as anticipated in the previous section.

4.1 Results for (GDLog, UCQ)

For a class $\mathbb{Q}$ of OMQs from (GDLog, UCQ), we denote with $\mathbb{D}_{\mathbb{Q}}$ the class of characteristic databases for OMQs from $\mathbb{Q}$ and with $\mathbb{Q}_c$ the class of covers for OMQs from $\mathbb{Q}$. The first main result which we show in this section is as follows:

▶ **Theorem 19** (Main Result 3). *Let $\mathbb{Q}$ be a r.e. enumerable class of OMQs from (GDLog, UCQ). Under the Exponential Time Hypothesis, the following are equivalent:*

1. *p-OMQ($\mathbb{Q}$) is fixed-parameter tractable*
2. *$\mathbb{Q}_c$ has bounded sub-modular width*
3. *$\mathbb{D}_{\mathbb{Q}}$ has bounded sub-modular width.*

Towards showing the result, the following fpt-reduction from evaluation of parameterized CSP to evaluation of parameterized OMQs from (GDLog, UCQ) plays an important role.

▶ **Theorem 20** (Main Result 2). *Let $\mathbb{Q}$ be a r.e. enumerable class of OMQs from (GDLog, UCQ). Then, there exists an fpt-reduction from p-CSP($\mathbb{D}_{\mathbb{Q}}$, _) to p-OMQ($\mathbb{Q}$).*

Proof. Let (D, B) be an instance of p-CSP($\mathbb{D}_{\mathbb{Q}}$, _). Also, let π be the projection mapping from $D \times B$ to D and let D_2 be the database obtained from $D \times B$ by dropping all atoms $R(\mathbf{a})$ for which $\pi|_{\mathbf{a}}$ is not injective. Then, $\pi|_{\mathsf{dom}(D_2)}$ is a homomorphism from D_2 to D which is i.g.s. As $D \in \mathbb{D}_{\mathbb{Q}}$, there exists an OMQ $Q \in \mathbb{Q}$, such that $D \in \mathbb{D}_Q$. Thus, there must some database D_0 which is qi w.r.t. Q, and a homomorphism $\uparrow$ from D to D_0 such that $(D, \uparrow) \in \mathsf{mdiv}(D_0, Q)$. Then $\pi\circ\uparrow$ is a homomorphism from D_2 to D_0 which is i.g.s. We can find Q and D_0 by enumeration, as all necessary tests can be decided using GSO encodings, similarly to the proof of Lemma 18. In the following, we denote with D^+ and D_2^+, the $\mathbf{S}$-databases $\mathsf{ext}(D, \uparrow, D_0) \in \mathbb{D}_Q^+$ and $\mathsf{ext}(D_2, \pi\circ\uparrow, D_0)$, resp. We will show that $D \to B$ iff $D_2^+ \models Q$. As $|D_2^+|$ is linear in $|B|$, we have an fpt-reduction.

'⇒': Assume that $D_2^+ \models Q$. The strategy of the proof is as follows: using Lemma 10 we construct a diversification (D_3, ι) of D_2 and then using the qi property of D_0, and subsequently of D^+, we show that $D_3 \preceq D$ and that $(D_3, g) \in \mathsf{div}(D_0, Q)$, for some homomorphism g. But $(D, \uparrow)$ is a minimal diversification of D_0 w.r.t. Q. Thus, $D \preceq D_3$, so $D \to D_3 \to D_2 \to D \times B \to B$.

Now to the details. Let p' be some contraction of a CQ p in Q which maps injectively only via an injective homomorphism h' into $\mathsf{ch}_{\mathcal{O}}(D_2^+)$ (denoted $p' \to^{io} \mathsf{ch}_{\mathcal{O}}(D_2^+)$) and let $A = \mathsf{ran}(h') \cap \mathsf{dom}(D_2)$. Also, let π^+ be the extension of the projection homomorphism π from D_2 to D to a homomorphism from D_2^+ to D^+. Then, π^+ is a homomorphism also from $\mathsf{ch}_{\mathcal{O}}(D_2^+)$ to $\mathsf{ch}_{\mathcal{O}}(D^+)$ and $\pi^+ \circ h'$ is a homomorphism from p' to $\mathsf{ch}_{\mathcal{O}}(D^+)$. As D_0 is qi w.r.t Q, according to Lemma 4, so is D^+. Also, as $p' \to^{io} \mathsf{ch}_{\mathcal{O}}(D_2^+)$, and $D_2^+ \to D^+$, it must be the case that p' maps injectively ony into $\mathsf{ch}_{\mathcal{O}}(D^+)$ ($p' \to^{io} \mathsf{ch}_{\mathcal{O}}(D^+)$). Otherwise, there would be some contraction p'' of p' such that $p'' \to \mathsf{ch}_{\mathcal{O}}(D^+)$ and $p'' \not\to \mathsf{ch}_{\mathcal{O}}(D_2^+)$ which would violate the fact that D^+ is qi w.r.t. Q. Thus, $\pi^+ \circ h'$ is injective.

Further on, as $A \subseteq \mathsf{ran}(h')$, it follows that $\pi^+_{|A}$ is injective. Also, as $A \subseteq \mathsf{dom}(D_2)$, it follows that $\pi_{|A}$ is injective.

As D_2^+ is of the form $\mathsf{ext}(D_2, \pi\circ\uparrow, D_0)$, according to Lemma 10 there exists a diversification (D_3, ι) of D_2 such that ι is the identity on $\mathsf{ker}(D_3)$, $\mathsf{ker}(\mathsf{D}_3) \subseteq A$, and $\mathsf{ext}(D_3, g, D_0) \models Q$, where g is the composition homomorphism $\uparrow \circ\pi \circ \iota$ from D_3 to D_0. Let D_3^+ be $\mathsf{ext}(D_3, g, D_0)$. Note that g is i.g.s. Also, let $h = \pi \circ \iota$ be the composition homomorphism from D_3 to D. It can be seen that h is also i.g.s. Figure 1 provides an overview of all the homomorphisms employed in the proof: the markings "io" on arrows denote the existence of injective only homomorphisms. Note that while the figure depicts extended databases, e.g. databases like $\mathsf{ext}(D, \uparrow, D_0)$ (D^+), most of the depicted homomorphisms are between the "base" databases, e.g. D_3, D_2, D, D_0.

We show next that $h|_{\mathsf{ker}(D_3)}$ is also injective. We have that $\mathsf{ran}(\iota_{|\mathsf{ker}(\mathsf{D}_3)}) = \mathsf{ker}(\mathsf{D}_3)$, thus $h|_{\mathsf{ker}(D_3)} = \pi \circ \iota_{|\mathsf{ker}(\mathsf{D}_3)}$ is the same as $\pi_{|\mathsf{ker}(\mathsf{D}_3)} \circ \iota_{|\mathsf{ker}(\mathsf{D}_3)}$. As $\mathsf{ker}(\mathsf{D}_3) \subseteq A$, $\pi_{|A}$ is injective, and $\iota_{|\mathsf{ker}(\mathsf{D}_3)}$ is the identity function, it follows that $h|_{\mathsf{ker}(D_3)}$ is injective. Recall that h is also i.g.s.

$$\begin{tikzcd}
p \arrow[r] \arrow[dr] & p' \arrow[r, "h'\ \mathrm{io}"] \arrow[rr, bend left, "\mathrm{io}"] & \mathsf{ch}_{\mathcal{O}}(D_2^+) \arrow[r, "\pi", "\text{i.g.s.}"'] & \mathsf{ch}_{\mathcal{O}}(D^+) \arrow[r, "\uparrow", "\text{i.g.s.}"'] & \mathsf{ch}_{\mathcal{O}}(D_0) \\
 & \mathsf{ch}_{\mathcal{O}}(D_3^+) \arrow[ur, bend right, "\iota", "\text{i.g.s.}"'] \arrow[urr, bend right, "h", "\text{i.g.s.}"'] \arrow[urrr, bend right, "g", "\text{i.g.s.}"'] & & &
\end{tikzcd}$$

Figure 1 Constructions for Proof of Direction $\Rightarrow$ of Theorem 20.

Thus, (D_3, h) is a diversification of D and $D_3 \preceq D$. As $D \preceq D_0$, it follows that $D_3 \preceq D_0$. At the same time, $\mathsf{ext}(D_3, g, D_0) \models Q$, thus $(D_3, g) \in \mathsf{div}D_0, Q$. But $(D, \uparrow) \in \mathsf{mdiv}(D_0, Q)$, thus $D \preceq D_3$. Thus, $D \to D_3 \to D_2 \to D \times B \to B$.

'$\Leftarrow$': Assume that $D \to B$. Then, D maps into $D \times B$ via some homomorphism h. Let $A = \mathsf{ran}(h)$. At the same time, $D \times B$ maps into D via the projection mapping π. Thus, $\pi \circ h$ is a homomorphism from D to itself. As D is a core, $\pi \circ h$ must be injective, and thus $\pi|_A$ is injective. Then, the database $(D \times B)|_A$ is a sub-structure of D_2, the structure obtained from $D \times B$ by removing all facts $r(\mathbf{a})$ for which $\pi|_{\mathbf{a}}$ is not injective. Thus, h is a homomorphism from D to D_2, or $D^+ \to D_2^+$. As $D^+ \models Q$, it follows that $D_2^+ \models Q$. ◀

We now prove the counter-positive of direction "1 $\Rightarrow$ 3" of Theorem 19. Let us assume that $\mathbb{D}_{\mathbb{Q}}$ has unbounded submodular width. As all $D \in \mathbb{D}_{\mathbb{Q}}$ are cores and cores witness semantic submodular width [16], $\mathbb{D}_{\mathbb{Q}}$ must have unbounded semantic submodular width. As explained in the Introduction, the following is known about evaluating uniform CSPs:

▶ **Theorem 21** (Theorem 1, [16]). *Let $\mathbb{C}$ be a recursively enumerable class of structures. Assuming the Exponential Time Hypothesis, p-CSP$(\mathbb{C}, _)$ is fixed-parameter tractable if and only if $\mathbb{C}$ has bounded semantic submodular width.*

Then, according to Theorem 21, p-CSP$(\mathbb{D}_{\mathbb{Q}}, _)$ is not fpt, and based on our reduction from Theorem 20, p-OMQ$(\mathbb{Q})$ is not fpt either.

To establish direction "3 $\Rightarrow$ 2" of Theorem 19, we show the following:

▶ **Lemma 22.** *Let $Q = (\mathcal{O}, \mathbf{S}, q)$ be an OMQ from $(GDLog, UCQ)$ and $Q_c = (\mathcal{O}, \mathbf{S}, q_c)$ be its cover. Then, $\mathsf{SMW}(Q_c) \leq \mathsf{SMW}(\mathbb{D}_Q)$.*

Proof Sketch. We show that for every CQ p_c in q_c there exists a database in $\mathbb{D}_Q$ of higher or equal submodular width. Every such p_c is of the form $h(p) \cup S$, where h is a homomorphism from a CQ p in q to $\mathsf{ch}_{\mathcal{O}}(D^+)$ for some $D^+ \in \mathbb{D}_Q^+$ and S is a set of atoms such that $D \subseteq S \subseteq D^+$ and every atom $r(\mathbf{a})$ from $S \setminus D$ has the property that $\mathbf{a}$ is a maximal guarded set in D^+. The latter follows from the minimality requirement on S in Definition 14 as well as from Point (2b) of the same definition. As every atom from $h(p)$ is guarded by some atom in S, it follows that $\mathsf{SMW}(p_c) = \mathsf{SMW}(S)$. Then, by careful manipulations of tree decompositions on D, D^+, and S, it can be further shown that: $\mathsf{SMW}(S) \leq \mathsf{SMW}(D)$. ◀

We next show the upper bound, i.e. direction "2 $\Rightarrow$ 1" of Theorem 19: if $\mathbb{Q}_c$ has bounded SMW, then, p-OMQ$(\mathbb{Q})$ is fixed-parameter tractable. The result follows from the fact that for an OMQ Q, Q_c is computable from Q (Lemma 18) and the following lemma (which concerns also OMQs based on GTGD):

▶ **Lemma 23.** *Let Q be an OMQs from (GTGD, UCQ) of bounded submodular width. Then, p-OMQ(Q) is fixed-parameter tractable.*

Proof. We use some results from [3]. Let L be the language of *linear tgds*, i.e. tgds which contain only one atom in the body. For an OMQ $Q = (\mathcal{O}, \mathbf{S}, q)$ from (GTGD, UCQ), and a database D, Lemma A.3 from [3], shows how to construct in FPT another OMQ $(\mathcal{O}^*, \mathbf{S}^*, q)$ from (L, UCQ) and a database D^* such that $D \models Q$ iff $D^* \models Q^*$. Then, Lemma A.1 from [3] shows that deciding whether $D^* \models Q^*$ can be done by considering a finite portion of $\mathsf{ch}_{\mathcal{O}^*}(D^*)$, which again can be computed in FPT. Thus, $D \models Q$ iff $D' \models q$, where D' is a database which can be computed in FPT. Thus, assuming that q has bounded submodular width, according to Theorem 21, deciding whether $D' \models q$ is in FPT as well. ◀

4.2 Results for (GTGD, UCQ)

The main result which we show in this section is as follows.

▶ **Theorem 24** (Main Result 1)**.** *Let $\mathbb{Q}$ be a r.e. class of OMQs from (GTGD, UCQ). Under the Exponential Time Hypothesis, $\mathbb{Q}$ has bounded semantic submodular width iff p-OMQ($\mathbb{Q}$) is fixed-parameter tractable.*

To show the result, we exploit the fact that for every OMQ $Q \in (\text{GTGD}, \text{UCQ})$, there exists an OMQ $Q' \in (\text{GDLog}, \text{UCQ})$, *the existential rewriting of* Q, such that $Q \equiv Q'$ [3]. Let $\mathbb{Q}'$ be the corresponding class of existential rewritings of OMQs from $\mathbb{Q}$.

Then, if p-OMQ($\mathbb{Q}$) is fpt, so is p-OMQ($\mathbb{Q}'$) and according to Theorem 19, the class of covers of OMQs from $\mathbb{Q}'$, $\mathbb{Q}'_c$, has bounded submodular width. But, $\mathbb{Q} \equiv \mathbb{Q}'_c$, and thus, $\mathbb{Q}$ has bounded semantic submodular width.

We turn our attention to the other direction of the theorem. In this case there exists a class of OMQs $\mathbb{Q}_k$ from (GTGD, UCQ) of bounded submodular width, such that $\mathbb{Q} \equiv \mathbb{Q}_k$. According to Lemma 23, p-OMQ($\mathbb{Q}_k$) is fpt, but we do not know how to compute $\mathbb{Q}_k$. The same holds about the class of its existential rewritings $\mathbb{Q}'_k$ from (GDLog, UCQ): it can be evaluated in fpt, but it is not given a priori. From Theorem 19 we know that the set of characteristic databases of $\mathbb{Q}'_k$, $\mathbb{D}_{\mathbb{Q}'_k}$ must have bounded submodular width.

A natural question is whether the notions used for characterizing OMQs from (GDLog, UCQ) are semantic, e.g. do equivalent OMQs have the same set of characteristic databases? If that would be the case, as $\mathbb{Q}' \equiv \mathbb{Q}'_k$, the set of characteristic databases $\mathbb{D}_{\mathbb{Q}'}$ would be identical to $\mathbb{D}_{\mathbb{Q}'_k}$, and thus would have bounded submodular width. Then, according to Theorem 19, p-OMQ($\mathbb{Q}'$) is fpt and so is p-OMQ($\mathbb{Q}$). However, this is not the case:

▶ **Example 25.** Let $Q_1 = (\mathcal{O}_1, \mathbf{S}, q_1)$ and $Q_2 = (\mathcal{O}_2, \mathbf{S}, q_2)$ be the following two OMQs:

$$\begin{array}{lll} \mathcal{O}_1 = \{R(x,y) \to A(x)\} & \mathbf{S} = \{R\} & q_1 = \exists x\ A(x) \\ \mathcal{O}_2 = \emptyset & \mathbf{S} = \{R\} & q_2 = \exists x, y\ R(x,y) \end{array}$$

Also, let $D = \{R(a,a)\}$ be an $\mathbf{S}$-database. It can be checked that $Q_1 \equiv Q_2$, $D \models Q_1$, and $D \models Q_2$. However, D is qi w.r.t. Q_1, but not w.r.t. Q_2 and $D \in \mathbb{D}_{Q_1}$, but $D \notin \mathbb{D}_{Q_2}$.

Still, when considering two equivalent OMQs Q_1 and Q_2 from (GDLog, UCQ), it is possible to construct another equivalent OMQ Q_{12} based on the intersection of the sets of (extended) characteristic databases of the two OMQs, $\mathbb{D}_{Q_1}$ and $\mathbb{D}_{Q_2}$. This can be done by starting with Q_1 (or Q_2 for that matter) and following the process described in Section 3 to construct the cover $Q_{1,c}$ of Q_1, except that in this case we use $\mathbb{D}_{Q_1} \cap \mathbb{D}_{Q_2}$ instead of $\mathbb{D}_{Q_1}$.

The new OMQ $Q_{12} = (\mathcal{O}, \mathbf{S}, q_{12})$ has the property that every CQ in q_{12} is a CQ in $q_{1,c}$, the UCQ from the cover $Q_{1,c}$ of Q_1. Thus $Q_{12} \subseteq Q_{1,c}$. At the same time, $Q_1 \subseteq Q_{12}$, and thus $Q_1 \equiv Q_{12}$. The submodular width of q_{12} will be bounded by the submodular width of $\mathbb{D}_{Q_1} \cap \mathbb{D}_{Q_2}$. Thus, if the submodular width of $\mathbb{D}_{Q_2}$ is bounded by some $k > 0$, there exists an OMQ $Q_{12} \equiv Q_1$ of submodular width bounded by k which can be constructed by removing all CQs with submodular width greater than k from the cover of Q_1. This allows us to prove direction "⇒" of Theorem 24: given the existence of (GDLog, UCQ) fpt witnesses $\mathbb{Q}'_k$ as above, we can construct from $\mathbb{Q}'$ a new class of (GDLog, UCQ) OMQs $\mathbb{Q}''$ of bounded submodular width which is equivalent to $\mathbb{Q}'$ and thus also to $\mathbb{Q}$. Thus, p-OMQ($\mathbb{Q}$) is fpt.

Finally, we obtain as a corollary of Theorem 19 and Theorem 24 the following result concerning covers which was already anticipated in Section 3:

▶ **Corollary 26.** *Let $\mathbb{Q}$ be a class of OMQs from (GDLog, UCQ). Then, $\mathbb{Q}$ has bounded semantic submodular width iff its class of covers $\mathbb{Q}_c$ has bounded submodular width.*

The question remains open whether covers are witnesses for semantic submodular width when considered individually, as opposed to witnesses for bounded semantic submodular width when considered as classes, as established in Corollary 26. We leave this open for now.

5 Revisiting the Bounded Arity Case

We revisit the characterization for fpt evaluation of OMQs from (GTGD, UCQ) over bounded arity schemas from [3]. A class of OMQs $\mathbb{Q}$ is *over bounded arity schemas* if there exists r such that every schema in some OMQ in $\mathbb{Q}$ contains only symbols of arity at most r.

▶ **Theorem 27** (Theorem 5.3, [3]). *Let $\mathbb{Q}$ be a r.e. class of OMQs from (GTGD, UCQ) over bounded arity schemas. Assumming $\mathrm{FPT} \neq \mathrm{W}[1]$, the following are equivalent :*

1. *p-OMQ($\mathbb{Q}$) is fixed-paramater tractable.*
2. *$\mathbb{Q}$ has bounded semantic treewidth.*

If either statement is false, then p-OMQ($\mathbb{Q}$) is $\mathrm{W}[1]$-hard.

The characterization generalizes a previous result concerning complexity of evaluating OMQs from ($\mathcal{ELHI}_\bot$, UCQ) [4]. At the same time, it can be seen as a generalization of Grohe's complexity results regarding the parameterized complexity of uniform CSPs over bounded arity schemas [24]:

▶ **Theorem 28** (Theorem 1, [24]). *Assume that $\mathrm{FPT} \neq \mathrm{W}[1]$. Then for every r.e. class $\mathbb{C}$ of structures of bounded arity the following statements are equivalent:*

1. *CSP($\mathbb{C}$, _) is in polynomial time.*
2. *p-CSP($\mathbb{C}$, _) is fixed-paramater tractable.*
3. *$\mathbb{C}$ has bounded treewidth modulo homomorphic equivalence.*

If either statement is false, then p-CSP($\mathbb{C}$, _) is $\mathrm{W}[1]$-hard.

Direction '2 ⇒ 1' of Theorem 27 is established in [3] using arguments regarding the chase construction and applying the result from [24] on a finite portion of the chase. As concerns the lower bound (direction '1 ⇒ 2'), the proof from [3] is extremely complex: it lifts in a non-trivial way the fpt-reduction from the parameterized k-clique problem to parameterized uniform CSPs over bounded arity schemas from [24]. Lifting the reduction to the OMQ case required in particular introspection into a certain construction used in the original proof, *the Grohe database*, and modifying that construction.

Here we sketch how it is possible to establish the results from Theorem 27 using Grohe's result as a black box by employing the reduction from Theorem 20. This shows the potential of the reduction for lifting results from the CQ evaluation realm to the OMQ one. We start by establishing a counterpart of Main Result 3 for the case of bounded arities OMQs.

▶ **Theorem 29** (GDLog Bounded Arity Characterization). *Let $\mathbb{Q}$ be a r.e. class of OMQs from (GDLog, UCQ) over bounded arity schemas. Assuming that* $\mathrm{FPT} \neq \mathrm{W}[1]$*:*

1. *p-OMQ($\mathbb{Q}$) is fixed-parameter tractable iff*
2. *$\mathbb{Q}_c$ has bounded tree-width iff*
3. *$\mathbb{D}_{\mathbb{Q}}$ has bounded tree-width.*

If either statement is false, then p-CSP($\mathbb{C}$, _) is W[1]*-hard.*

The strategy to prove Theorem 29 is similar to the one used to prove Theorem 19, except that this time Theorem 28 from [24] is used as a blackbox, as opposed to the unbounded arity case, where the results for uniform CSPs over unbounded arity schemas from Theorem 21 were used as a blackbox. This is the case both for showing the upper bound, direction '2 ⇒ 1' of the theorem, and the lower bound, direction '1 ⇒ 3' of the theorem. In the latter case, we use again our reduction from parameterized uniform CSPs to parameterized OMQs. What still needs to be shown, is the connection between the treewidth of the cover of an OMQ and the treewidth of the set of characteristic databases of the same OMQ. It follows straightaway from the proof of Lemma 22 that:

▶ **Lemma 30.** *For $Q = (\mathcal{O}, \mathbf{S}, q)$ an OMQ from $(GDLog, UCQ)$ with schema arity at most r, and $Q_c = (\mathcal{O}, \mathbf{S}, q_c)$ its cover, it is the case that $\mathsf{TW}(Q_c) \leq \mathsf{max}(r, \mathsf{TW}(\mathbb{D}_Q))$.*

Direction '3 ⇒ 2' of Theorem 29 follows from Lemma 30. By using Theorem 29, we can then retrieve the results from Theorem 27 similarly as we did for Theorem 24.

6 Conclusions and Future Work

In this work, we characterized the fpt border for evaluating classes of parameterized OMQs based on guarded TGDs and UCQs in the unbounded arity case. For ontologies expressed in GDLog, we provided a characterization based on new constructions, namely sets of characteristic databases and covers of an OMQ. On the way to establish this result, we introduced an fpt reduction from evaluating parameterized uniform CSPs to evaluating parameterized OMQs.

The reduction enables the lifting of results from the CSP world to the OMQ one in a modular fashion. To further showcase this, we revisited the case of OMQs over bounded arity schemas, previously addresed in [3]. For classes of such OMQs from (GDLog, UCQ) we established a new syntactic characterization of the tractability border in terms of covers and characteristic databases, while for classes of OMQs from (GTGD, UCQ), we showed how the reduction enabled a much simpler proof for the semantic characterization from [3].

Here we only considered Boolean OMQs, as the corresponding results for CQ evaluation in the unbounded arity case have also only been established in the Boolean case. As future work, we plan to extend our work to the non-Boolean case. Many results from the database world concerning efficiency of performing tasks like counting [17], enumeration [7, 14], and so on, use structural measures on the queries similar to the ones involved in the characterizations for evaluation. Thus, by generalizing our constructs to the non-Boolean case, depending on which structural measures are preserved when transitioning from sets of characteristic databases to covers of OMQs, we might be able to lift such results to the OMQ world.

References

1 Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*, 2003.

2 Jean-François Baget, Michel Leclère, Marie-Laure Mugnier, and Eric Salvat. On rules with existential variables: Walking the decidability line. *Artif. Intell.*, 175(9-10):1620–1654, 2011.

3 Pablo Barceló, Victor Dalmau, Cristina Feier, Carsten Lutz, and Andreas Pieris. The limits of efficiency for open- and closed-world query evaluation under guarded tgds. In *PODS*, pages 259–270, 2020.

4 Pablo Barceló, Cristina Feier, Carsten Lutz, and Andreas Pieris. When is ontology-mediated querying efficient? In *LICS*, pages 1–13, 2019.

5 Pablo Barceló, Diego Figueira, Georg Gottlob, and Andreas Pieris. Semantic optimization of conjunctive queries. *J. ACM*, 67(6):34:1–34:60, 2020.

6 Pablo Barceló, Andreas Pieris, and Miguel Romero. Semantic optimization in tractable classes of conjunctive queries. *SIGMOD Rec.*, 46(2):5–17, 2017.

7 Christoph Berkholz and Nicole Schweikardt. Constant delay enumeration with fpt-preprocessing for conjunctive queries of bounded submodular width. In *MFCS 2019*, volume 138, pages 58:1–58:15, 2019.

8 Meghyn Bienvenu, Balder ten Cate, Carsten Lutz, and Frank Wolter. Ontology-based data access: A study through disjunctive datalog, csp, and MMSNP. *ACM Trans. Database Syst.*, 39(4):33:1–33:44, 2014.

9 Achim Blumensath. Guarded second-order logic, spanning trees, and network flows. *Logical Methods in Computer Science*, 6(1), 2010.

10 Pierre Bourhis, Marco Manna, Michael Morak, and Andreas Pieris. Guarded-based disjunctive tuple-generating dependencies. *ACM Trans. Database Syst.*, 41(4):27:1–27:45, 2016.

11 Andrea Calì, Georg Gottlob, and Michael Kifer. Taming the infinite chase: Query answering under expressive relational constraints. *J. Artif. Intell. Res.*, 48:115–174, 2013.

12 Andrea Calì, Georg Gottlob, and Thomas Lukasiewicz. Datalog±: a unified approach to ontologies and integrity constraints. In *ICDT*, pages 14–30, 2009.

13 Andrea Calì, Georg Gottlob, and Andreas Pieris. Towards more expressive ontology languages: The query answering problem. *Artif. Intell.*, 193:87–128, 2012.

14 Nofar Carmeli and Markus Kröll. Enumeration Complexity of Conjunctive Queries with Functional Dependencies. In *(ICDT 2018)*, volume 98, pages 11:1–11:17, 2018.

15 Chandra Chekuri and Anand Rajaraman. Conjunctive query containment revisited. *Theor. Comput. Sci.*, 239(2):211–229, 2000.

16 Hubie Chen, Georg Gottlob, Matthias Lanzinger, and Reinhard Pichler. Semantic width and the fixed-parameter tractability of constraint satisfaction problems. In Christian Bessiere, editor, *IJCAI*, pages 1726–1733, 2020.

17 Hubie Chen and Stefan Mengel. A Trichotomy in the Complexity of Counting Answers to Conjunctive Queries. In *(ICDT 2015)*, volume 31, pages 110–126, 2015.

18 Bruno Courcelle. The monadic second-order logic of graphs XIV: uniformly sparse graphs and edge set quantifications. *Theor. Comput. Sci.*, 299(1-3):1–36, 2003.

19 Alin Deutsch, Alan Nash, and Jeff B. Remmel. The chase revisisted. In *PODS*, pages 149–158, 2008.

20 Rodney G. Downey and Michael R. Fellows. Fixed-parameter tractability and completeness I: basic results. *SIAM Journal on Computing*, 24:873–921, 1995.

21 Jan Foniok. Homomorphisms and structural properties of relational systems, 2007. `arXiv:0710.4477`.

22 Erich Grädel. Decision procedures for guarded logics. In *CADE*, volume 1632, 1999.

23 Erich Grädel, Colin Hirsch, and Martin Otto. Back and forth between guarded and modal logics. *ACM Trans. Comput. Log.*, 3(3):418–463, 2002.

24 Martin Grohe. The complexity of homomorphism and constraint satisfaction problems seen from the other side. *J. ACM*, 54(1):1:1–1:24, 2007.

25 Stijn Heymans, Jos de Bruijn, Livia Predoiu, Cristina Feier, and Davy Van Nieuwenborgh. Guarded hybrid knowledge bases. *Theory Pract. Log. Program.*, 8(3):411–429, 2008.

26 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001.

27 David S. Johnson and Anthony C. Klug. Testing containment of conjunctive queries under functional and inclusion dependencies. *J. Comput. Syst. Sci.*, 28(1):167–189, 1984.

28 Vladimir Lifschitz. Action languages, answer sets, and planning. In *APT*, pages 357–374, 1999.

29 David Maier, Alberto O. Mendelzon, and Yehoshua Sagiv. Testing implications of data dependencies. *ACM Trans. Database Syst.*, 4(4):455–469, 1979.

30 Dániel Marx. Tractable hypergraph properties for constraint satisfaction and conjunctive queries. In *STOC*, pages 735–744, 2010.

31 Riccardo Rosati. On the decidability and complexity of integrating ontologies and rules. *J. Web Semant.*, 3(1):41–60, 2005.

32 Mihalis Yannakakis. Algorithms for acyclic database schemes. In *VLDB*, pages 82–94, 1981.