Existential Second-Order Logic over Graphs: Parameterized Complexity

Max Bannach
European Space Agency, Advanced Concepts Team, Noordwijk, The Netherlands

Florian Chudigiewitsch
Universität zu Lübeck, Germany

Till Tantau
Universität zu Lübeck, Germany

Abstract

By Fagin's Theorem, NP contains precisely those problems that can be described by formulas starting with an existential second-order quantifier, followed by only first-order quantifiers (ESO formulas). Subsequent research refined this result, culminating in powerful theorems that characterize for each possible sequence of first-order quantifiers how difficult the described problem can be. We transfer this line of inquiry to the *parameterized* setting, where the size of the set quantified by the second-order quantifier is the parameter. Many natural parameterized problems can be described in this way using simple sequences of first-order quantifiers: For the clique or vertex cover problems, two universal first-order quantifiers suffice ("for all pairs of vertices ... must hold"); for the dominating set problem, a universal followed by an existential quantifier suffice ("for all vertices, there is a vertex such that ..."); and so on. We present a complete characterization that states for each possible sequence of first-order quantifiers how high the parameterized complexity of the described problems can be. The uncovered dividing line between quantifier sequences that lead to tractable versus intractable problems is distinct from that known from the classical setting, and it depends on whether the parameter is a lower bound on, an upper bound on, or equal to the size of the quantified set.

2012 ACM Subject Classification Theory of computation → Finite Model Theory; Theory of computation → Complexity theory and logic; Theory of computation → Fixed parameter tractability; Theory of computation → W hierarchy

Keywords and phrases existential second-order logic, graph problems, parallel algorithms, fixed-parameter tractability, descriptive complexity

Digital Object Identifier 10.4230/LIPIcs.IPEC.2023.3

Related Version *Full Version*: https://arxiv.org/abs/2310.01134 [2]

Acknowledgements We thank Marcel Wienöbst for fruitful discussions and helpful comments on an earlier draft.

1 Introduction

The 3-coloring problem is to decide, given an undirected simple graph, whether there exist three sets R, G, and B (the red, green, and blue vertices) such that any two vertices x and y connected by an edge have different colors; or in logical terms:

$$\exists R \exists G \exists B\, \forall x \forall y \Big((Rx \lor Gx \lor Bx) \land \big(x \sim y \to \neg\big((Rx \land Ry) \lor (Gx \land Gy) \lor (Bx \land By)\big)\big)\Big). \quad (1)$$

This formula is an *existential second-order* formula, meaning that it starts with existential second-order quantifiers ($\exists R \exists G \exists B$) followed by first-order quantifiers ($\forall x \forall y$) followed by a quantifier-free part. We can succinctly describe which quantifiers are used in such a prefix

18th International Symposium on Parameterized and Exact Computation (IPEC 2023).
Editors: Neeldhara Misra and Magnus Wahlström; Article No. 3; pp. 3:1–3:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

by using "E_1" for a (monadic, hence the "$_1$") existential second-order quantifier and "e" and "a" for existential and universal first-order quantifiers, respectively. The resulting *quantifier pattern* of the above formula is then $E_1E_1E_1aa$; and (monadic) existential second-order formulas are formulas with a prefix in $E_1^*(ae)^*$. It is no coincidence that an NP-complete problem can be described using the quantifier pattern $E_1E_1E_1aa$: Fagin's Theorem [12] states that a problem lies in NP iff it can be described by a formula with a pattern in $E_i^*(ae)^*$ for some arity i. However, the example shows that the pattern $E_1E_1E_1aa$ already suffices to describe an NP-complete problem and a closer look reveals that so does E_1E_1aa. In contrast, formulas with the pattern E_iaa can only describe problems decidable in NL (regardless of the arity i of the quantified relation variables). Such observations have sparked an interest in different quantification patterns' power. The question was answered by Gottlob, Kolaitis, and Schwentick [15] in the form of a dichotomy ("can only describe problems in P" versus "can describe an NP-complete problem") and later in a refined form by Tantau [17], where the described problems in P are further classified into "in AC^0" or "L-complete" or "NL-complete".

While in the formula for 3-colorability it was only necessary that three sets of colors *exist,* for many problems the *size* of these sets is important. Consider:

$$\phi_{\text{clique}} = \exists^{\geq} C\,\forall x \forall y\big((Cx \wedge Cy) \to x \sim y\big), \tag{2}$$

$$\phi_{\text{vertex-cover}} = \exists^{\leq} C\,\forall x \forall y\big(x \sim y \to (Cx \vee Cy)\big), \tag{3}$$

$$\phi_{\text{dominating-set}} = \exists^{\leq} D\,\forall x \exists y\big(Dy \wedge (x = y \vee x \sim y)\big). \tag{4}$$

where the second-order quantifiers $\exists^{\geq}$ and $\exists^{\leq}$ ask whether there exists a set of size at least or at most some parameter value k such that the rest of the formula holds. These formulas show that we can describe the clique problem using a formula with the succinctly written pattern $E_1^{\geq}aa$ (and also $E_1^{=}aa$); the vertex cover problem using $E_1^{\leq}aa$ (and again also $E_1^{=}aa$); and the dominating set using $E_1^{\geq}ae$ (and yet again also $E_1^{=}ae$). Readers will notice that the problems are some of the most fundamental problems studied in that theory and lie in different levels of the W-hierarchy. The main message of the present paper is that it is once more *no coincidence that the quantifier patterns needed to describe these problems differ* ($E_1^{\geq}aa$ versus $E_1^{\leq}aa$ versus $E_1^{\geq}ae$): As done in [15, 17], we will give a complete characterization of the complexities of the problems that can be described using a specific quantifier pattern. The well-known results that the (parameterized) clique and dominating set problems are W[1]-hard while the (parameterized) vertex cover problem lies in FPT (in fact, in para-AC^0) can now all be derived just from the syntactic structure of the formulas used to describe these problems.

Our Contributions. In this paper, we classify the complexity of the following classes (formal definitions are given in Section 2): Given a pattern $p \in \{a, e\}^*$ of first-order quantifiers, the classes p-FD($E_1^{=}p$), p-FD($E_1^{\leq}p$), and p-FD($E_1^{\geq}p$) contain all parameterized problems that can be described by formulas with quantifier pattern $E_1^{=}p$, or $E_1^{\geq}p$, or $E_1^{\leq}p$, respectively. The restriction to study just a *single, monadic, parameterized* ESO quantifier is motivated by our earlier observation that important and interesting problems of parameterized complexity can be described in this way. Our classification is complete in the sense that for every p we either show that all problems in the class are fixed-parameter tractable (in these cases, we derive more fine-grained results by placing the problems in para-AC^0 or para-$\mathrm{AC}^{0\uparrow}$) or there is a W[1]-hard problem that can be described using the pattern. Table 1 lists the obtained bounds. In the table, the classes with the subscript "basic" refer to the restriction to undirected graphs without self-loops. As can be seen, for these graphs we get slightly

Table 1 Complete complexity classification of the weighted ESO logic for a single weighted monadic second-order quantification followed by first-order quantifiers with some pattern $p \in \{a,e\}^*$ (where $p \preceq q$ means that p is a subsequence of q). The upper part (arbitrary structures) and lower part (basic graphs) are identical except for the patterns $E_1^{\geq}ae$ and $E_1^{\leq}aa$, where they differ. Note that $\text{para-AC}^0 \subsetneq \text{para-AC}^{0\uparrow} \subseteq \text{para-P} = \text{FPT}$ and $\text{FPT} \cap \text{W[1]-hard} = \emptyset$ is a standard assumption.

$\text{p-FD}(E_1^{=}p)$	$\subseteq$ para-AC^0, when	$p \preceq e^*a$.
	$\cap$ W[1]-hard $\neq \emptyset$, when	ae or $aa \preceq p$.
$\text{p-FD}(E_1^{\geq}p)$	$\subseteq$ para-AC^0, when	$p \preceq e^*a$.
	$\cap$ W[1]-hard $\neq \emptyset$, when	ae or $aa \preceq p$.
$\text{p-FD}(E_1^{\leq}p)$	$\subseteq$ para-AC^0, when	$p \preceq e^*a$.
	$\nsubseteq$ para-AC^0 but $\subseteq$ para-AC$^{0\uparrow}$, when	$aa \preceq p \preceq e^*a^*$.
	$\cap$ W[1]-hard $\neq \emptyset$, when	$ae \preceq p$.
$\text{p-FD}_{\text{basic}}(E_1^{=}p)$	$\subseteq$ para-AC^0, when	$p \preceq e^*a$.
	$\cap$ W[1]-hard $\neq \emptyset$, when	ae or $aa \preceq p$.
$\text{p-FD}_{\text{basic}}(E_1^{\geq}p)$	$\subseteq$ para-AC^0, when	$p \preceq e^*a$ or ae.
	$\cap$ W[1]-hard $\neq \emptyset$, when	aee, eae, or $aa \preceq p$.
$\text{p-FD}_{\text{basic}}(E_1^{\leq}p)$	$\subseteq$ para-AC^0, when	$p \preceq e^*a$ or aa.
	$\nsubseteq$ para-AC^0 but $\subseteq$ para-AC$^{0\uparrow}$, when	$aaa \preceq p \preceq e^*a^*$.
	$\cap$ W[1]-hard $\neq \emptyset$, when	$ae \preceq p$.

different complexity results. This is in keeping with the classical, non-parameterized setting studied by Gottlob et al. [15], where results for basic graphs are often *considerably* harder to obtain. However, the complexity landscape we uncover in the present paper is different from the one presented in [15] and [17]: Although certain patterns (like $p = ae$) feature prominently in the parameterized and non-parameterized analysis, the dividing lines are different. To establish these lines, we combine ideas used in the classical setting with different methods from parameterized complexity theory, tailored to the specific problems we study. The notoriously difficult cases from the classical setting (Gottlob et al. [15] spend 34 pages to address the case E_1^*ae, Tantau [17] spends several pages on E_1aa) are also technically highly challenging in the parameterized setting.

Our research sheds new light on what difference it makes whether we want solutions to have size *exactly* k or *at most* k or *at least* k. To begin, equations (2) and (4) already show that for individual problems (like clique) the maximization problem can be hard while minimization is trivial (a single vertex is always a clique) and for some problems (like dominating set) the opposite is true (the whole vertex set itself is always a dominating set). Furthermore, from the perspective of descriptive complexity, there is a qualitative difference between $\exists^{=}C$ and $\exists^{\leq}C$ on the one hand and $\exists^{\geq}C$ on the other: For any k, the first two can easily be expressed in normal ESO logic using k first-order quantifiers binding the elements of C, while $\exists^{\geq}$ translates to $\exists C \exists x_1 \cdots \exists x_k$ where the x_i bind the elements not in C. Thus, $\exists^{=}$ and $\exists^{\leq}$ only allow us to express problems that are "slicewise first-order" and hence in $\text{XAC}^0 \subseteq \text{XP}$, while already the slice for $k = 0$ of $\exists^{\geq}$ formulas can express NP-complete problems for many patterns. *However,* we also prove a result for basic graphs for $p = ae$ that runs counter this "tendency" of $\exists^{\geq}$ to be harder than $\exists^{\leq}$: While $\text{p-FD}_{\text{basic}}(E_1^{\leq}ae)$ contains the W[2]-hard dominating set problem, $\text{p-FD}_{\text{basic}}(E_1^{\geq}ae) \subseteq \text{para-AC}^0$.

Related Work. Using logic to describe languages dates back all the way to Büchi's pioneering work [6] on the expressive power of monadic second-order logic (which, over strings, describes exactly the regular languages). Switching from monadic second-order logic to existential second-order logic yields Fagin's Theorem [12]. Since then, the expressive power of fragments of this logic was the subject of intensive research: Eiter et al. [11] studied the expressiveness of ESO-patterns over strings; Gottlob et al. did so over graphs [15]; Tantau [17] refined the latter results for subclasses of P. Taken together, these results give us a complete complexity-theoretic classification of the problems resulting from any ESO quantifier pattern over strings, basic graphs, directed graphs, undirected graphs, and arbitrary structures.

Using logical fragments to characterize complexity classes is also standard practice in parameterized complexity theory [13], especially the power of MSO logic plays a prominent role, see for instance [8]. In particular, characterizations of the levels of the W-hierarchy in terms of the number of quantifier alternations are known [9, 13], but – to the best of our knowledge – a complete and exact analysis of the parameterized complexity of problems in terms of the quantifier patterns describing them is new.

Organization of this Paper. Following a review of basic concepts and terminology in Section 2, we present our results on the power of quantifier patterns of the forms $E_1^{\leq}p$, $E_1^{\geq}p$, and $E_1^{=}p$ for $p \in \{a,e\}^*$ in the subsections of Section 3 (arbitrary structures) and Section 4 (basic graphs). Due to space constraints, we provide only proof ideas for some of the results. The full proofs can be found in the technical report version [2] of the text.

2 Background in Descriptive and Parameterized Complexity

Terminology for Graphs and Logic. A *directed graph* ("digraph") is a pair $G = (V, E)$ where V is a *vertex set* and $E \subseteq V \times V$ an *edge set*. An *undirected graph* is a pair $G = (V, E)$ such that $E \subseteq \big\{\{u, v\} \mid u, v \in V\big\}$. A *basic graph* is an undirected graph that has no self-loops, that is, where all edges have size 2. In this paper, graphs are always finite.

We use standard terminology from logic and finite model theory, see for instance [10]. Let us point out some perhaps not-quite-standard notation choices: Our vocabularies τ (also known as *signatures*) contain *only relation symbols* and we write $\textsc{struc}[\tau]$ to denote the set of all finite τ-structures. For a first-order or second-order τ-sentence ϕ (a formula without free variables), let $\textsc{models}(\phi)$ denote the subset of $\textsc{struc}[\tau]$ of all τ-structures that are models of ϕ. As an example, we can represent digraphs using the vocabulary $\tau_{\text{digraph}} = \{\sim^2\}$, containing a single binary relation symbol, and the class of digraphs is exactly $\textsc{struc}[\tau_{\text{digraphs}}]$. The formula $\phi = \forall x \forall y(x \sim y \to x \neq y)$ expresses that there are no loops in a graph, that is, $\textsc{models}(\phi) = \{G \mid G \text{ is a digraph that has no self-loops}\}$. While an undirected graph $G = (V, E)$ is not immediately a τ_{digraph}-structure, we can trivially "turn it" into a structure $\mathcal{G}$ by setting the universe to be V and setting $\sim^{\mathcal{G}} = \{(x, y) \mid \{x, y\} \in E\}$ and this structure is a model of $\phi_{\text{undirected}} = \forall x \forall y(x \sim y \to y \sim x)$. The structures representing basic graphs are then models of $\phi_{\text{basic}} = \forall x \forall y(x \sim y \to (x \neq y \land y \sim x))$. As another example, the class of all bipartite graphs equals $\textsc{models}(\phi_{\text{bipartite}})$ where $\phi_{\text{bipartite}}$ is the second-order formula $\exists X \forall u \forall v\big(u \sim v \to (Xu \leftrightarrow \neg Xv)\big)$ and Xu is our shorthand for the less concise $X(u)$.

As already sketched in the introduction, we can associate a *quantifier prefix pattern* (a word over the infinite alphabet $\{E_1, E_2, E_3, \dots\} \cup \{e, a\}$), or just a *pattern*, to formulas of ESO logic by first writing them in prenex normal form (quantifiers first, in a block) and then replacing each (existential) second-order quantifier by E_i, where i is the arity of the quantifier, each universal first-order quantifier by a, and each existential first-order quantifier by e. For instance, the pattern of $\phi_{\text{bipartite}}$ is E_1aa.

Describing Problems and Classes. In the context of descriptive complexity a *decision problem* P is a subset of $\text{STRUC}[\tau]$ that is closed under isomorphisms. We say that ϕ *describes* P if $\text{MODELS}(\phi) = P$. Moving on to classes, for a set Φ of τ-formulas, let $\text{FD}(\Phi) := \{\text{MODELS}(\phi) \mid \phi \in \Phi\}$ denote the class of problems "Fagin-defined" by Φ. For a quantifier prefix pattern p let $\text{FD}(p) := \{\text{MODELS}(\phi) \mid \phi \text{ has pattern } p\}$, so (1) shows that $3\text{COLORABLE} \in \text{FD}(E_1E_1E_1aa)$, and for a set S of patterns let $\text{FD}(S) = \bigcup_{p\in S} \text{FD}(p)$. In slight abuse of notation, we usually write down regular expressions to denote sets S of quantifier patterns: For instance, Fagin's Theorem [12] can now be written as "$\text{NP} = \text{FD}\big(E_2^*(ae)^*\big)$." Trivially, more quantifiers potentially allow us to express more problems. Formally, let $p \preceq q$ denote that p is a subsequence of q (so p can be obtained from q by, possibly, deleting some letters). Then $\text{FD}(p) \subseteq \text{FD}(q)$. Also in slight abuse of notation, we also write things like "$p \preceq e^*a$" to indicate that $p \preceq q$ holds for some q of the form e^*a.

Our analysis will show that restricting attention to basic graphs yields particularly interesting results. For this reason, it will be convenient to consider the introduced complexity classes restricted to basic graphs by adding a subscript "basic": For τ_{digraph}-formulas ϕ, let $\text{MODELS}_{\text{basic}}(\phi) = \text{MODELS}(\phi) \cap \text{MODELS}(\phi_{\text{basic}})$ and define $\text{FD}_{\text{basic}}(\Phi)$ and $\text{FD}_{\text{basic}}(p)$ in the obvious ways – and similarly for the classes with the subscript "undirected."

When we move from classical complexity theory to parameterized complexity, we assign to every *instance* a *parameter* that measures an aspect of interest of that instance and that is hopefully small for practical instances. A *parameterized problem* is a set $Q \subseteq \text{STRUC}[\tau] \times \mathbb{N}$ such that for every k the *slice* $\{x \mid (x, k) \in Q\}$ is closed under isomorphisms. In a pair (x, k) we call x the *input* and k the *parameter*. The usual goal in the field is to prove that a problem is *fixed-parameter tractable* (in FPT) by deciding $(x, k) \in^? Q$ in time $f(k) \cdot |x|^{O(1)}$ for some computable function f. In the context of problems described by ESO formulas, a natural parameter to consider is the *size of the relations that we can assign to the existential second-order quantifiers* and this size is commonly called the *solution weight*. As mentioned earlier, problems like the vertex cover problem can be described naturally in this manner: Consider the formula $\phi(X) = \forall u \forall v\big(u \sim v \to (Xu \vee Xv)\big)$, where X is a free monadic second-order variable. Then for a graph $G = (V, E)$, viewed as a logical structure $\mathcal{G}$, and a set $C \subseteq V$ we have $\mathcal{G} \models \phi(C)$ iff C is a vertex cover of G. Thus, $(\mathcal{G}, k) \in \text{p-VERTEX-COVER} = \{(\mathcal{G}, k) \mid \mathcal{G}$ has a vertex cover of size $k\}$ iff there exists a set $C \subseteq V$ of size k such that $\mathcal{G} \models \phi(C)$.

Formally, the second-order quantifiers $\exists^{\leq}$, $\exists^{=}$, and $\exists^{\geq}$ have the following semantics: For a structure $\mathcal{S}$ with a universe U, a non-negative integer k, an i-ary second-order variable X, and a formula $\phi(X)$, we say that $\mathcal{S}$ *is a model of* $\exists^{\leq} X\, \phi(X)$ *for parameter* k and write $(\mathcal{S}, k) \models \exists^{\leq} X\, \phi(X)$, if there is a set $C \subseteq U^i$ with $|C| \leq k$ such that $\mathcal{S} \models \phi(C)$. A formula starting with a $\exists^{\leq}$ quantifier then gives rise to a parameterized problem: Let $\text{p-MODELS}\big(\exists^{\leq} X\, \phi(X)\big) = \{(\mathcal{S}, k) \mid (\mathcal{S}, k) \models \exists^{\leq} X\, \phi(X)\}$ and let $\text{p-FD}(\Phi) := \{\text{p-MODELS}(\phi) \mid \phi \in \Phi\}$. The at-least and equal cases are, of course, defined analogously. As an example, we have $\text{p-VERTEX-COVER} \in \text{p-FD}(E_1^{\leq}aa)$ since $\text{p-VERTEX-COVER} = \text{p-MODELS}\Big(\exists^{\leq} X\, \forall u \forall v\big(u \sim v \to (Xu \vee Xv)\big)\Big)$.

Standard and Parameterized Complexity Classes. Concerning standard complexity classes, we use standard definitions, see for instance [1, 16]. In the context of descriptive complexity theory, it is often necessary to address coding issues (meaning the question of how words are encoded as logical structures and *vice versa*) – but fortunately this will not be important for the present paper. Concerning parameterized complexity classes like FPT = para-P or W[1], we also use standard definitions, which can be adapted to the descriptive setting in exactly the same way as for classical complexity classes (see for instance [3, 4] for details) and encoding details will once more be unimportant. The classes para-AC^0 and para-$\mathrm{AC}^{0\uparrow}$ are likely less well-known: We have $Q \in$ para-AC^0 if there is a family $(C_{n,k})_{n,k\in\mathbb{N}}$ of unbounded fan-in circuits of constant depth and size $f(k) \cdot n^{O(1)}$ for some computable function f, such that for every $(\mathcal{S}, k) \in \textsc{struc}[\tau] \times \mathbb{N}$ we have $(\mathcal{S}, k) \in Q$ iff the circuit $C_{\mathrm{length}(\mathcal{S}),k}$ outputs 1 on input of (a suitably encoded) $\mathcal{S}$, where length$(\mathcal{S})$ is the length of the encoding of $\mathcal{S}$. For the class para-$\mathrm{AC}^{0\uparrow}$, the circuits may have depth $f(k)$. We have para-$\mathrm{AC}^0 \subsetneq$ para-$\mathrm{AC}^{0\uparrow} \subseteq$ para-P = FPT [3]. In our proofs, two properties of the classes will be important: First, all of them are (quite trivially) closed under para-AC^0-reductions. Second, for $\tau = (I^1)$, the signature with a single unary relation symbol, we have p-THRESHOLD $= \{(\mathcal{S}, k) \mid \mathcal{S} = (U, I^{\mathcal{S}}), |I^{\mathcal{S}}| \geq k\} \in$ para-AC^0, that is, we can "count up to the parameter in para-AC^0." For more details on these classes, including discussions of uniformity, see [3, 4, 7].

3 Classifying Parameterized ESO Classes: Arbitrary Structures

We now begin tracing the tractability frontier for the classes from the upper part of Table 1: p-FD$(E_1^{=}p)$, p-FD$(E_1^{\geq}p)$, and p-FD$(E_1^{\leq}p)$. Recall that for these classes we are given a formula ϕ starting with one of the monadic second-order quantifiers $\exists^{=}$, $\exists^{\leq}$, or $\exists^{\geq}$, followed by first-order quantifiers with the pattern p; and the objective is to show *upper bounds* of the form "for all ϕ with pattern p all p-MODELS(ϕ) lie in a certain class" and *lower bounds* of the form "there is a ϕ with pattern p such that p-MODELS(ϕ) contains a problem that is hard for a certain class". We dedicate one subsection to each of $\exists^{=}$, $\exists^{\leq}$, and $\exists^{\geq}$, each starting with the main theorem and covering more technical parts of the proofs later.

In this section, we allow arbitrary (finite, relational) structures, meaning that the signature τ can contain arbitrary relation symbols (but neither constant nor function symbols), and our upper bounds will hold for all such structures. However, for our *lower bounds* it will suffice to consider only *undirected graphs.* That is, the lower bounds for a pattern p will be of the form "there is a ϕ with pattern p such that p-MODELS$_{\mathrm{undirected}}(\phi)$ contains a hard problem". Interestingly, we can typically (*but not always,* by the results of Section 4) replace *undirected* graphs by *basic* graphs (undirected graphs without self-loops) here.

3.1 Solution Weight Equals the Parameter for Arbitrary Structures

We start with the classification of p-FD$(E_1^{=}p)$, the first two lines of Table 1:

▶ **Theorem 3.1** (Complexity Dichotomy for p-FD$(E_1^{=}p)$). *Let $p \in \{a, e\}^*$ be a pattern.*
1. p-FD$(E_1^{=}p) \subseteq$ para-AC^0, *if $p \preceq e^*a$.*
2. p-FD$(E_1^{=}p)$ *contains a* W[1]*-hard problem, if $aa \preceq p$ or $ae \preceq p$.*

Both items also hold for p-FD$_{\mathrm{undirected}}(E_1^{=}p)$ *and even* p-FD$_{\mathrm{basic}}(E_1^{=}p)$.

The cases in the above theorem are exhaustive (so for every p we either have $p \preceq e^*a$ or we have $aa \preceq p$ or $ae \preceq p$). The theorem follows directly from the following lemma:

▶ **Lemma 3.2** (Detailed Bounds for p-FD$(E_1^{=}p)$).
1. p-FD$(E_1^{=}e^*a) \subseteq$ para-AC^0.
2. p-FD$_{\mathrm{basic}}(E_1^{=}aa)$ *contains a* W[1]*-hard problem.*
3. p-FD$_{\mathrm{basic}}(E_1^{=}ae)$ *contains a* W[2]*-hard problem.*

Proof. Item 1 is shown in Corollary 3.5, which we prove later in this section. For item 2, we already saw in equation (2) that we can describe the W[1]-hard clique problem p-CLIQUE using a formula ϕ_{clique} with pattern $E_1^{\geq} aa$. It was also already mentioned that replacing $\exists^{\geq}$ by $\exists^{=}$ yields the same problem and, thus, p-$\text{FD}_{\text{basic}}(E_1^{=} aa)$ contains a W[1]-hard problem. Similarly, for item 3, replacing $\exists^{\leq}$ by $\exists^{=}$ in equation (4) shows we can describe the W[2]-hard dominating set problem using an $E_1^{=} ae$ formula. ◀

To establish the upper bound (item 1 of the theorem), we make use of a well-known connection between weighted satisfiability in predicate logic (problems in p-FD($E_1^{=} e^* a$) in our case) and weighted satisfiability in propositional logic (the problem p-1WSAT$^{=}$ below). We present this connection in more generality than strictly necessary to prove the upper bound since we will rely on variants of it later on. For propositional formulas ψ in dCNF (meaning *at most* d literals per clause), let $\text{vars}(\psi)$ and $\text{clauses}(\psi)$ denote the sets of variables and clauses, respectively. For an *assignment* $\beta\colon \text{vars}(\psi) \to \{0,1\}$, with the model relation $\beta \models \psi$ defined as usual, the *weight* is $\text{weight}(\beta) = \bigl|\{v \in \text{vars}(\psi) \mid \beta(v) = 1\}\bigr|$. The following problem is the weighted version of the satisfiability problem for dCNF formulas:

▶ **Problem 3.3** (p-dWSAT$^{=}$ for fixed d).
Instance: *A* dCNF *formula* ψ *and a non-negative integer* $k \in \mathbb{N}$.
Parameter: k
Question: *Is there an assignment* β *with* $\beta \models \psi$ *and* $\text{weight}(\beta) = k$*?*

The problem is also known as p-dWSAT in the literature, but we keep the "$=$" superscript since we also consider p-dWSAT$^{\leq}$ and p-dWSAT$^{\geq}$, where we ask whether there is a satisfying assignment with $\text{weight}(\beta) \leq k$ and $\text{weight}(\beta) \geq k$, respectively. For us, the importance of these problems lies in the following lemma, where "$\leq_{\text{para-AC}^0}$" refers to the earlier-mentioned para-AC^0-reductions. Recall that these reductions are extremely weak and that all classes considered in this paper, including para-AC^0, are closed under them.

▶ **Lemma 3.4.** *Let* $d \geq 1$*. Then:*
1. *For every* $Q \in$ p-FD($E_1^{=} e^* a^d$) *we have* $Q \leq_{\text{para-AC}^0}$ p-dWSAT$^{=}$*.*
2. *For every* $Q \in$ p-FD($E_1^{\geq} e^* a^d$) *we have* $Q \leq_{\text{para-AC}^0}$ p-dWSAT$^{\geq}$*.*
3. *For every* $Q \in$ p-FD($E_1^{\leq} e^* a^d$) *we have* $Q \leq_{\text{para-AC}^0}$ p-dWSAT$^{\leq}$*.*

Proof. In all three cases, Q is the set of models of a weighted ESO formula of the form $\exists^{=} X\, \phi(X)$ or $\exists^{\leq} X\, \phi(X)$ or $\exists^{\geq} X\, \phi(X)$ where $\phi(X)$ has the quantifier pattern $e^* a^d$. In [13, Lemma 7.2] it is shown that given a formula $\phi(X)$ with such a pattern, we can map any structure $\mathcal{S}$ with some universe S to a dCNF formula ψ such that there there is a one-to-one correspondence between the sets $C \subseteq S$ with $\mathcal{S} \models \phi(C)$ and the satisfying assignments β of ψ. Furthermore, when C corresponds to β, we have $|C| = \text{weight}(\beta)$. While in [13] it is only argued that the mapping from $\phi(X)$ to ψ can be done in polynomial time, a closer look reveals that a para-AC^0 reduction suffices. This means that in all three items we can use this mapping as the reduction whose existence in claimed. ◀

▶ **Corollary 3.5.** p-FD($E_1^{=} e^* a$)*,* p-FD($E_1^{\leq} e^* a$)*,* p-FD($E_1^{\geq} e^* a$) *are subsets of* para-AC^0*.*

Proof. Let us start with some $Q \in$ p-FD($E_1^{=} e^* a$). By item 1 of Lemma 3.4, $Q \leq_{\text{para-AC}^0}$ p-1WSAT$^{=}$. Thus, showing p-1WSAT$^{=} \in$ para-AC^0 yields the claim for p-FD($E_1^{=} e^* a$) as para-AC^0 is closed under para-AC^0 reductions. However, a 1CNF formula ψ is just a conjunction of literals. It is trivial to check in plain AC^0 (independently of the parameter) whether ψ is satisfiable (it may not contain a literal and its negation) and, if so, it is trivial to determine the single satisfying assignment $\beta\colon \text{vars}(\psi) \to \{0,1\}$. We are left with having to check whether weight$(\beta) = k$ holds. It is well known [3] that the problem of checking whether the number of 1 bits in a bitstring is at least, at most, or equal to a parameter value lies in para-AC^0, yielding the claim. However, this also yields the other two items. ◂

3.2 Solution Weight Is At Least the Parameter for Arbitrary Structures

For p-FD$(E_1^{\geq}p)$, we get the exact same dichotomy as for p-FD$(E_1^{=}p)$. However, a look at the detailed bounds in the lemma shows that for p-FD$(E_1^{\geq}p)$ we only get a lower bound for *undirected* graphs and not for *basic* graphs (and, indeed, we will show in Section 4 that the complexity is different for basic graphs). Furthermore, while we always have p-FD$(E_1^{=}p) \subseteq \mathrm{W}[t]$ for some t (see [13, Definition 5.1]), we show that the patterns $E_1^{\geq}eae$ or $E_1^{\geq}aee$ suffice to describe even para-NP-complete problems even on basic graphs. Thus, although the tractability frontier ("in FPT" versus "contains W[1]-hard problems") is the same for p-FD$(E_1^{=}p)$ and p-FD$(E_1^{\geq}p)$, the detailed structure is more complex.

▸ **Theorem 3.6** (Complexity Dichotomy for p-FD$(E_1^{\geq}p)$). *Let p be a pattern.*
1. p-FD$(E_1^{\geq}p) \subseteq$ para-AC^0*, if* $p \preceq e^*a$*.*
2. p-FD$(E_1^{\geq}p)$ *contains a* W[1]*-hard problem, if* $aa \preceq p$ *or* $ae \preceq p$*.*

Both items also hold for p-FD$_{\text{undirected}}(E_1^{\geq}p)$*.*

The theorem follows directly from the following lemma (whose last two items are not actually needed here, but shed more light on the detailed structure and *will* be needed in Section 4).

▸ **Lemma 3.7** (Detailed Bounds for p-FD$(E_1^{\geq}p)$).
1. p-FD$(E_1^{\geq}e^*a) \subseteq$ para-AC^0.
2. p-FD$_{\text{basic}}(E_1^{\geq}aa)$ *contains a* W[1]*-hard problem.*
3. p-FD$_{\text{undirected}}(E_1^{\geq}ae)$ *contains a* para-NP*-hard problem.*
4. p-FD$_{\text{basic}}(E_1^{\geq}eae)$ *contains a* para-NP*-hard problem.*
5. p-FD$_{\text{basic}}(E_1^{\geq}aee)$ *contains a* para-NP*-hard problem.*

Proof. Item 1 is already stated in Corollary 3.5. For item 2, equation 2 shows that the W[1]-complete problem p-CLIQUE can be expressed with a weighted ESO formula with the pattern $E_1^{\geq}aa$ and, thus, lies in p-FD$_{\text{basic}}(E_1^{\geq}aa)$.

For the other items, a claim is useful:

▹ Claim 3.8. If there is an NP-hard problem in FD(E_1p), there is a para-NP-hard problem in p-FD$(E_1^{\geq}p)$; and this holds also for the restrictions to undirected, basic, or directed graphs.

To see that this claim holds, just note that the non-parameterized problem is the special case of the parameterized maximization problem where $k = 0$.

To prove item 3, observe that Gottlob et al. have shown [15, Theorem 2.1] that there are NP-complete problems in FD$_{\text{undirected}}(E_1ae)$. By the claim, there must be para-NP-hard problems in p-FD$_{\text{undirected}}(E_1^{\geq}ae)$. Next, for item 4, in [15, Theorem 2.5] it is shown that there is an NP-hard problem in FD$_{\text{basic}}(E_1eae)$. Finally, for item 5, by [15, Theorem 2.6] there is also an NP-hard problem in FD$_{\text{basic}}(E_1aee)$. ◂

Note that compared to Lemma 3.2, in the third item of Lemma 3.7 we have shown the lower bound for the restriction of structures to undirected graphs rather than basic graphs. This is no coincidence: the self-loops which are allowed for undirected graphs have in some cases an impact on the complexity of the problems we can express. Later, when we cover basic graphs, we will see that, indeed, sometimes problems become easier compared to their counterparts where undirected graphs are admissible as structures.

3.3 Solution Weight Is At Most the Parameter for Arbitrary Structures

When the parameter is an upper bound on the weight of solutions, the tractability landscape changes quite a bit: The pattern $E_1^{\leq}ae$ becomes intractable, while $E_1^{\leq}aa$ no longer lies in para-AC^0, but stays tractable:

▶ **Theorem 3.9** (Complexity Trichotomy for $\text{p-FD}(E_1^{\leq}p)$). *Let p be a pattern.*
1. $\text{p-FD}(E_1^{\leq}p) \subseteq \text{para-AC}^0$*, if* $p \preceq e^*a$*.*
2. $\text{p-FD}(E_1^{\leq}p) \subseteq \text{para-AC}^{0\uparrow}$ *but* $\text{p-FD}(E_1^{\leq}p) \nsubseteq \text{para-AC}^0$*, if* $aa \preceq p \preceq e^*a^*$*.*
3. $\text{p-FD}(E_1^{\leq}p)$ *contains a* $\mathrm{W}[1]$*-hard problem, if* $ae \preceq p$*.*

All items also hold for $\text{p-FD}_{\text{undirected}}(E_1^{\leq}p)$*.*

As before, the theorem covers all possible p and follows from a lemma that is a bit more general than strictly necessary: We will need items 4 and 5 only in Section 4.2, where we show that item 3 does *not* hold for basic graphs.

▶ **Lemma 3.10** (Detailed Bounds for $\text{p-FD}(E_1^{\leq}p)$).
1. $\text{p-FD}(E_1^{\leq}e^*a) \subseteq \text{para-AC}^0$.
2. $\text{p-FD}(E_1^{\leq}e^*a^*) \subseteq \text{para-AC}^{0\uparrow}$.
3. $\text{p-FD}_{\text{undirected}}(E_1^{\leq}aa)$ *contains a problem not in* para-AC^0*.*
4. $\text{p-FD}_{\text{basic}}(E_1^{\leq}aaa)$ *contains a problem not in* para-AC^0*.*
5. $\text{p-FD}_{\text{basic}}(E_1^{\leq}eaa)$ *contains a problem not in* para-AC^0*.*
6. $\text{p-FD}_{\text{basic}}(E_1^{\leq}ae)$ *contains a* $\mathrm{W}[2]$*-hard problem.*

Proof. Item 1 is already stated in Corollary 3.5. Item 2 is shown in Lemma 3.11 below. Items 3, 4, and 5 are shown in Lemma 3.14, Lemma 3.15, and Lemma 3.16, respectively. Item 6 follows, once more, from p-DOMINATING-SET $\in \text{p-FD}_{\text{basic}}(E_1^{\leq}ae)$ by equation (4). ◀

▶ **Lemma 3.11.** $\text{p-FD}(E_1^{\leq}e^*a^*) \subseteq \text{para-AC}^{0\uparrow}$.

Proof. Let $Q \in \text{p-FD}(E_1^{\leq}e^*a^d)$ for some fixed d. By Lemma 3.4, $Q \leq_{\text{para-AC}^0} \text{p-}d\text{WSAT}^{\leq}$. We now show that $\text{p-}d\text{WSAT}^{\leq} \in \text{para-AC}^{0\uparrow}$, which implies the claim as $\text{para-AC}^{0\uparrow}$ is closed under para-AC^0 reductions.

We have to construct a circuit family of depth $f(k)$ and size $f(k) \cdot n^{O(1)}$ for $n = |\text{vars}(\psi)|$ for some computable function f. The circuit implements a bounded search tree such that every layer evaluates one level of the tree. To that end, each layer gets a set Ψ_i of formulas as input and outputs a new set Ψ_{i+1} of formulas. We start with $\Psi_0 = \{\psi\}$. The invariant will be that ψ has a satisfying assignment of (exact) weight w iff some formula in Ψ_i has a satisfying assignment of (exact) weight $w - i$.

To compute the next Ψ_{i+1} for $i \in \{0, \dots, k\}$, we perform the following operations in parallel for every $\rho \in \Psi$:
1. If every clause in ρ contains a negative literal (meaning that ρ is satisfied by the all-0 assignment), accept the original input. (Doing so is correct by the invariant.)
2. Take a clause $c \in \text{clauses}(\rho)$ that only contains positive literals $x_1, \dots, x_e$. For each x_i, generate a new formula ρ^i from ρ by "setting one of these variables to 1" or, formally, by removing all clauses that contain it positively and removing the variable from all clauses that contain it negatively, respectively. Add ρ^1 to ρ^e to Ψ_{i+1}. (This maintains the invariant as we *must* set one of the x_i to 1 in any assignment that satisfies ρ.)

If we have not accepted the input after having computed Ψ_{k+1}, we reject. This is correct since all satisfying assignments of the $\rho \in \Psi_{k+1}$ have weight at least 0 and, thus, by the invariant all satisfying assignments of ψ have weight at least $k + 1 - 0 > k$.

To see that the circuit can be implemented with the claimed depth and size, note that since $e \leq d$, the list grows by a factor of at most d in every layer and we can implement each layer in constant depth. As there are only $k+1$ layers, we have $|\Psi_{k+1}| \leq (k+1)^d =: f(k)$. ◀

For the three remaining still-to-be-proved lower bounds in Lemma 3.10, the claim is always that a class is (unconditionally) not contained in para-AC^0. To prove this, we will show that the following problem is (provably) not in para-AC^0 but can be para-AC^0-reduced to problems that lie in the three classes:

▶ **Problem 3.12** (p-MATCHED-REACH).

Instance: *A directed layered graph G with vertex set $\{1,\dots,n\} \times \{1,\dots,k\}$, where the ith* layer *is $V_i := \{1,\dots,n\} \times \{i\}$, such that for each $i \in \{1,\dots,k-1\}$ the edges point to the next layer and they form a perfect matching between V_i and V_{i+1}; two designated vertices $s \in V_1$ and $t \in V_k$.*
Parameter: *k.*
Question: *Is t reachable from s?*

▶ **Lemma 3.13.** p-MATCHED-REACH $\notin$ para-AC^0 *and consequently, for any problem Q with* p-MATCHED-REACH $\leq_{\text{para-AC}^0}$ *Q we have $Q \notin$* para-AC^0.

Proof. Beame et al. [5] have shown that any depth-c circuit that solves p-MATCHED-REACH requires size $n^{\Omega(k^{(\rho^{-2c})/3})}$, where ρ is the golden ratio. However, p-MATCHED-REACH $\in$ para-AC^0 would imply that for some constant c there is a depth-c circuit family that decides the problem in size $f(k) \cdot n^{O(1)}$; contradicting the Beame et al. bound of $n^{k^{\Theta(1)}}$. For the claim concerning Q, just note that para-AC^0 is closed under para-AC^0 reductions. ◀

▶ **Lemma 3.14.** p-$\text{FD}_{\text{undirected}}(E_1^{\leq}aa) \not\subseteq$ para-AC^0.

Proof. Consider the following formula with quantifier pattern $E_1^{\leq}aa$:

$$\phi_{\text{reach}} := \exists^{\leq} S\, \forall x \forall y \big(\big((x \sim x) \to Sx\big) \land \big((Sx \land x \sim y) \to Sy\big)\big).$$

We claim that we can reduce p-MATCHED-REACH to p-$\text{MODELS}_{\text{undirected}}(\phi_{\text{reach}})$ as follows (and the claim then follows from Lemma 3.13): On input (G, s, t), the reduction first checks that the graph is, indeed, a layered graph with perfect matchings between consecutive levels. Then, we forget about the direction of the edges (making the graph undirected). Next, we add an additional layer V_{k+1} and match each vertex of V_k to the corresponding new vertex V_{k+1}. Next, we *remove* the just-added edge from t in layer V_k to its counterpart in layer V_{k+1}. Finally, add a self-loop at s. To see that this reduction is correct, note that the self-loop at s forces it (but does not *force* any other vertex), to be part of the solution set S by the first part of the formula. The second part then forces the solution set to be closed under reachability. Thus, if t lies on the same path as s, there is a solution of size k, and if not, the smallest solution has size $k+1$. ◀

▶ **Lemma 3.15.** p-$\text{FD}_{\text{basic}}(E_1^{\leq}aaa) \not\subseteq$ para-AC^0.

Proof. We reduce p-MATCHED-REACH to p-$\text{MODELS}_{\text{basic}}(\phi_{\text{reach-aaa}})$ for

$$\phi_{\text{reach-aaa}} := \exists^{\leq} S\, \forall x \forall y \forall z \big(((x \sim y \land y \sim z \land x \sim z) \to Sx) \land ((Sx \land x \sim y) \to Sy)\big).$$

On input (G, s, t), once more we start by forgetting about the direction of the edges. This time, add two vertices and connect them to s so that these three vertices form a triangle. Do the same for t by adding another two vertices. Output $k+4$ as the new parameter. This reduction is correct, because the first part of the formula forces every vertex which is part of a triangle to be part of S, which are exactly the triangles at s and t. The latter part of the formula forces the solution set to be closed under reachability. Thus, if t lies on the same path as s, there is a solution of size $k+4$, and if not, the smallest solution has size as least $k+5$. ◀

▶ **Lemma 3.16.** p-$\mathrm{FD}_{\mathrm{basic}}(E_1^{\leq} eaa) \not\subseteq$ para-AC^0.

Proof. We reduce p-MATCHED-REACH to p-$\mathrm{MODELS}_{\mathrm{basic}}(\phi_{\text{reach-eaa}})$ for

$$\phi_{\text{reach-eaa}} := \exists^{\leq} S\, \exists z \forall x \forall y \big(Sz \wedge ((Sx \wedge x \sim y) \to Sy)\big).$$

On input (G, s, t) we forget the direction of edges and add a single vertex that we connect to every vertex that has degree 1 except for s and t. If t is on the same path as s, there will be two connected components: One consisting of the path between s and t, and one containing everything else. In particular, there is a component of size k and one of size $n-k$. If t is not on the same path as s, there is just a single connected component of size n. To see that the reduction is correct, notice that the first part of the formula (Sz) forces at least one vertex to be part of the solution. The latter part of the formula once more forces the solution set to be closed under reachability. By construction, there is a solution of size at most k iff t was on the same path as s. ◀

4 Classifying Parameterized ESO Classes: Basic Graphs

We saw in Section 3 that the parameterized complexity of weighted ESO classes depends strongly on the first-order quantifier pattern p and on whether we are interested in the equal-to, at-least, or at-most case – but it does *not* matter whether we consider arbitrary logical structures, only directed graphs, or only undirected graphs: the complexity is always the same. The situation changes if we restrict attention to *basic graphs*, which are undirected graphs without self-loops: We get different tractability frontiers. This is an interesting effect since the only difference between undirected graphs and basic graphs is that some vertices may have self-loops – and self-loops are usually neither needed nor used in hardness proofs, just think of the clique problem, the vertex cover problem, or the dominating set problem. Nevertheless, it turns out that "a single extra bit per vertex" and sometimes even "a single self-loop" allows us to encode harder problems than without.

To establish the tractability frontier for basic graphs, we can, of course, recycle many results from the previous section: Having a look at the detailed bounds listed in Lemmas 3.2, 3.7, and 3.10, we see that the upper bounds are shown for arbitrary structures and, hence, also hold for basic graphs; and many lower bounds have also already been established for basic graphs. Indeed, it turns out there are exactly two classes whose complexity "changes" when we restrict the inputs to basic graphs:

1. p-$\mathrm{FD}_{\mathrm{basic}}(E_1^{\geq} ae)$ lies in para-AC^0, while p-$\mathrm{FD}(E_1^{\geq} ae)$ does not.
2. p-$\mathrm{FD}_{\mathrm{basic}}(E_1^{\leq} aa)$ lies in para-AC^0, while p-$\mathrm{FD}(E_1^{\leq} aa)$ does not.

We have already shown the "while …" part in Section 3, it is the upper bounds that are new. For all other patterns p, the classification does not change. Proving the two items turns out to be technical and we devote one subsection to each of these results.

4.1 The Case $E_1^{\geq} ae$ for Basic Graphs

As mentioned, for the classification of the complexity of p-$\mathrm{FD}_{\mathrm{basic}}(E_1^{\geq} p)$ we can reuse all of our previous results, *except* that p-$\mathrm{FD}_{\mathrm{basic}}(E_1^{\geq} ae) \subseteq$ para-AC^0 holds. This is the statement of Lemma 4.2, which is proved in the rest of this section. However, before be plunge into the glorious details, let us ascertain that there are no further patterns $p \neq ae$ for which p-FD$_{\text{basic}}(E_1^{\geq} p)$ becomes any easier: To see this, note that for any p with $p \not\preceq ae$ we have $aa \preceq p$ or $eae \preceq p$ or $aee \preceq p$; and for aa, eae, and aee we have already established hardness for *basic* graphs in Lemma 3.7. For completeness, we spell out the resulting structure:

▶ **Theorem 4.1** (Dichotomy for p-FD$_{\text{basic}}(E_1^{\geq} p)$)**.** *Let p be a pattern.*

1. p-FD$_{\text{basic}}(E_1^{\geq} p) \subseteq$ para-AC^0*, if $p \preceq e^*a$ or $p \preceq ae$.*
2. p-FD$_{\text{basic}}(E_1^{\geq} p)$ *contains a* W[1]*-hard problem, if $aa \preceq p$, $eae \preceq p$, or $aee \preceq p$.*

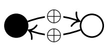
▶ **Lemma 4.2.** p-FD$_{\text{basic}}(E_1^{\geq} ae) \subseteq$ para-AC^0.

For the surprisingly difficult proof we employ machinery first used in [15] and in [17, Section 3.3]: Our objective is to represent the problems in p-FD$_{\text{basic}}(E_1^{\geq} ae)$ as special kinds of graph coloring problems – and to then show that we can solve these problems in para-AC^0.

Proof idea. Following [15], a *pattern graph* $P = (C, A^{\oplus}, A^{\ominus})$ consists of a set of *colors* C, a set $A^{\oplus} \subseteq C \times C$ of $\oplus$-arcs, and a set $A^{\ominus} \subseteq C \times C$ of $\ominus$-arcs (note that $A^{\oplus}$ and $A^{\ominus}$ need not be disjoint). In our paper, we will only need the case that there are only two colors, so $C = \{\text{black}, \text{white}\}$ will always hold, and we call such a pattern graph *binary.* In the rest of the section, *pattern* always refers to a *binary pattern graph* (and no longer to quantifier prenex patterns – we are only interested in the single pattern $E_1^{\geq} ae$ anyway). Observe that there 256 possible binary pattern graphs. A *superpattern* of a pattern $P = (C, A^{\oplus}, A^{\ominus})$ is any pattern $P' = (C, B^{\oplus}, B^{\ominus})$ with $A^{\oplus} \subseteq B^{\oplus}$ and $A^{\ominus} \subseteq B^{\ominus}$. A *$\oplus$-superpattern* is a superpattern with $A^{\ominus} = B^{\ominus}$.

For a basic graph $B = (V, E)$, a *coloring of* B is a function $c\colon V \to C$. However, unlike standard coloring problems, where vertices connected by an edge must have different colors, what constitutes an allowed coloring is dictated by the pattern graph via a *witness function:* A mapping $w\colon V \to V$ is called a *witness function for a coloring* c if for all $x \in V$ we have

1. $x \neq w(x)$,
2. if $\{x, w(x)\} \in E$, then $\big(c(x), c(w(x))\big) \in A^{\oplus}$, and
3. if $\{x, w(x)\} \notin E$, then $\big(c(x), c(w(x))\big) \in A^{\ominus}$.

The idea is that a vertex x and its witness $w(x)$ are connected by "a $\oplus$-arc" if there is an edge between them and by "a $\ominus$-arc" if there is no edge between them. The pattern graph then tells us which colors are allowed for x and $w(x)$ in dependence on which kind of arc there is. For instance, for the pattern  every vertex must be connected by an edge to a vertex of the opposite color. Note that this is not the same as asking for a 2-coloring: We only impose a requirement on the edge (corresponding to a $\oplus$-arc) between x and $w(x)$, other edges are not relevant. In more detail, consider a triangle with the vertices $\{x, y, z\}$ and the coloring $c(x) = \text{black}$, $c(y) = c(z) = \text{white}$ and the witness function $w(x) = y$ and $w(y) = w(z) = x$. Then the coloring is legal with respect to the pattern and the witness function, despite that fact that a triangle is not 2-colorable.

If there exists a coloring c together with a witness function w for B with respect to P, we say that *B is P-saturated by c and w.* The saturation problem SATURATION(P) for a pattern P is then simply the set of all basic graphs $B = (V, E)$ that can be P-saturated (via some coloring c and witness function w). The relation between the saturation problem and $E_1 ae$ is as follows: We want the witness function to tell us for each x in $\forall x$ which y in $\exists y$ we must pick to make a formula of the form $\exists S\, \forall x \exists y\, \psi$ true: *We color a vertex black to indicate that it should be included in S, otherwise we color it white.* In this way, one can associate a pattern graph with each $E_1 ae$-formula.

▶ **Fact 4.3** ([17, Fact 3.3] for a single quantifier). *For every* ESO *formula ϕ with quantifier pattern E_1ae there is a binary pattern graph P such that* MODELS$(\phi) =$ SATURATION(P).

(Strictly speaking, this only holds for basic graphs B with at least two vertices. For this reason, in the following we always assume that $|V| \geq 2$ holds.)

Adapting this approach to the parameterized setting is straightforward: Define the *weight* of a binary coloring as the number of vertices that are colored black. This leads to the following parameterized problem and transfer of Fact 4.3 to the parameterized setting:

▶ **Problem 4.4** (p-SATURATION$^{\geq}(P)$ for a fixed binary pattern graph $P = (C, A^{\oplus}, A^{\ominus})$).
Instance: *A basic graph $B = (V, E)$ and an integer $k \in \mathbb{N}$.*
Parameter: k.
Question: *Can B be P-saturated via a coloring of weight* at least k*?*

▷ Claim 4.5. For every weighted ESO formula ϕ with quantifier pattern $E_1^{\geq}ae$ there is a binary pattern graph P such that p-MODELS$(\phi) =$ p-SATURATION$^{\geq}(P)$.

With the above claim, it remains to show for all 256 binary pattern graphs P that p-SATURATION$^{\geq}(P) \in$ para-AC^0 holds. The (surprisingly diverse and nontrivial) treatment of the cases can be found in the technical report version [2]. ◀

4.2 The Case $E_1^{\leq}aa$ for Basic Graphs

The classification of the complexity of $E_1^{\leq}p$ also changes when we restrict the admissible input structures to be basic graphs: p-FD$_{\text{basic}}(E_1^{\leq}aa) \subseteq$ para-AC^0 holds by Lemma 4.7 and, once more, this is the only change. Proving the lemma will be considerably easier than in the previous section, but still demanding. Before we start, we summarize the resulting landscape for completeness. Note that all bounds other than the just-mentioned new upper bound have already been shown in Lemma 3.10.

▶ **Theorem 4.6** (Trichotomy for p-FD$_{\text{basic}}(E_1^{\leq}p)$). *Let p be a pattern.*
1. p-FD$_{\text{basic}}(E_1^{\leq}p) \subseteq$ para-AC^0 *if $p \preceq e^*a$ or $p \preceq aa$.*
2. p-FD$_{\text{basic}}(E_1^{\leq}p) \subseteq$ para-AC$^{0\uparrow}$ *but* p-FD$_{\text{basic}}(E_1^{\leq}p) \not\subseteq$ para-AC^0*, if $aaa \preceq p$ or $eaa \preceq p$, and $p \preceq e^*a^*$.*
3. p-FD$_{\text{basic}}(E_1^{\leq}p)$ *contains a* W[1]*-hard problem, if $ae \preceq p$.*

▶ **Lemma 4.7.** p-FD$_{\text{basic}}(E_1^{\leq}aa) \subseteq$ para-AC^0.

Proof idea. As in the previous section, we can reuse some ideas from the literature, but need to take care of some extra complications caused by the need to limit the sizes of the solution sets. In particular, we will use the notion of *cardinality constraints* introduced in [17] for the study of the E_1aa case: For two sets $C, D \subseteq \{0, 1, 2\}$ define p-CSP$^{\leq}\{C, D\}$ as follows. The instances for this problem consist of a finite universe U, a function P that maps *each* two-element subset $\{x, y\} \subseteq U$ to either C or D (so, unlike normal constraint satisfaction problems, a constraint must be stated for every single pair of variables), and a number k. A *solution* for P is a subset $X \subseteq U$ of size $|X| \leq k$ such that for all two-element subsets $\{x, y\} \subseteq U$ we have $|\{x, y\} \cap X| \in P(x, y)$. We call $\big(U, P^{-1}(C)\big)$ the *C-graph of P* and note that this is just the set of edges that are mapped to C by P. The *D-graph* is defined as $\big(U, P^{-1}(D)\big)$; and observe that every two-element set belongs to exactly one of these two graphs except when $C = D$ in which case both graphs are the complete cliques. It is shown in [17, Lemma 3.1] that all problems in FD(Eaa) reduce to CSP$\{C, D\}$ (without the "$\leq k$" restrictions) for some C and D. We need to following variant:

▷ Claim 4.8. All problems in p-FD($E_1^{\leq}aa$) reduce to p-CSP$^{\leq}\{C, D\}$ for some C and D via para-AC^0 reductions.

Proof. The reduction is a trivial reencoding in which the solutions of the CSP instances are exactly the sets that satisfy the formula when assigned to the existentially bound second-order variable. In particular, solutions and assigned sets have the same sizes and satisfy the same size restrictions. ◁

It remains to show p-CSP$^{\leq}\{C, D\} \in$ para-AC^0 for all $C, D \subseteq \{0, 1, 2\}$. For this, we have to go over the possible choices in a case distinction. Once more, the detailed treatment of the cases can be found in the technical report version [2]. ◀

5 Conclusion

We gave a complete characterization of the tractability frontier of weighted ESO logic over basic graphs, undirected graphs, and arbitrary structures. While in some cases our results mirror the classical complexity landscape, other cases yield clearly different results. The proofs differ significantly from the classical setting and make extensive use of tools from parameterized complexity theory. Especially for the class p-FD$_{\text{basic}}(E^{\geq}ae)$, sophisticated machinery is needed to establish the upper bound. Whether we require solutions to have size exactly k, at most k, or at least k plays a central role in the complexity of the describable problems. While the class p-FD$_{\text{basic}}(E^{\geq}ae)$ can be shown to be included in para-AC^0, the classes p-FD$_{\text{basic}}(E^{=}ae)$ and p-FD$_{\text{basic}}(E^{\leq}ae)$ both contain W[2]-hard problems. Similarly, while p-FD$_{\text{basic}}(E^{\leq}aa)$ is contained in para-AC^0, both p-FD$_{\text{basic}}(E^{=}aa)$ and p-FD$_{\text{basic}}(E^{\geq}aa)$ contain W[1]-hard problems.

An obvious further line of research is to consider the prefixes $E_i^{=}p$, $E_i^{\geq}p$, and $E_i^{\leq}p$ for $i \geq 2$, that is, the non-monadic case, and also multiple monadic quantifiers. While in the classical setting it turns out [15] that we can normally reduce non-monadic quantifiers to (possibly multiple) monadic ones, it is not clear whether the same happens in the parameterized setting. Just pinpointing the complexity of, say, p-FD$_{\text{basic}}(E_2^{\geq}ae)$ seems difficult.

A different line of inquiry is to further investigate the patterns that lead to intractable problems: In the unweighted setting, all ESO-definable problems lie in NP and if the class is not contained in P, then it contains an NP-complete problem. Our intractability results range from W[1]-completeness to para-NP-completeness. Can we find for every t a pattern for which we get classes that contain W[t]-hard problems and are contained in W[t]?

Our results also shed some light on *graph modification problems,* where we have a fixed first-order formula ϕ and are given a pair (G, k). The objective is to modify the graph as little as possible (for instance, by deleting as few vertices as possible) such that for the resulting graph G' we have $G' \models \phi$. Fomin et al. [14] have recently shown a complexity dichotomy regarding the number quantifier alternations in ϕ. Since it is not difficult to encode the "to be deleted vertices" using a $\exists^{\leq}D$ quantifier, at least the upper bounds from our paper also apply to vertex deletion problems. We believe that our results can be extended to also cover lower bounds and, thereby, to give exact and complete classifications of the complexity of vertex deletion problems in terms of the quantifier pattern of ϕ.

References

1 Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009.

2 Max Bannach, Florian Chudigiewitsch, and Till Tantau. Existential second-order logic over graphs: Parameterized complexity. Technical Report abs/2310.01134, Cornell University, 2023. URL: `https://arxiv.org/abs/2310.01134`.

3 Max Bannach, Christoph Stockhusen, and Till Tantau. Fast parallel fixed-parameter algorithms via color coding. In *10th International Symposium on Parameterized and Exact Computation, IPEC 2015, September 16-18, 2015, Patras, Greece*, pages 224–235, 2015. `doi:10.4230/LIPIcs.IPEC.2015.224`.

4 Max Bannach and Till Tantau. Computing kernels in parallel: Lower and upper bounds. In *13th International Symposium on Parameterized and Exact Computation, IPEC 2018, August 20-24, 2018, Helsinki, Finland*, pages 13:1–13:14, 2018. `doi:10.4230/LIPIcs.IPEC.2018.13`.

5 Paul Beame, Russell Impagliazzo, and Toniann Pitassi. Improved depth lower bounds for small distance connectivity. *Comput. Complex.*, 7(4):325–345, 1998. `doi:10.1007/s000370050014`.

6 J. Richard Büchi. Weak second-order arithmetic and finite automata. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 6(1-6):66–92, 1960.

7 Yijia Chen and Jörg Flum. Some lower bounds in parameterized ac^0. In *41st International Symposium on Mathematical Foundations of Computer Science, MFCS 2016, August 22-26, 2016 - Kraków, Poland*, pages 27:1–27:14, 2016. `doi:10.4230/LIPIcs.MFCS.2016.27`.

8 Bruno Courcelle and Joost Engelfriet. *Graph Structure and Monadic Second-Order Logic - A Language-Theoretic Approach*, volume 138 of *Encyclopedia of mathematics and its applications*. Cambridge University Press, 2012. URL: `http://www.cambridge.org/fr/knowledge/isbn/item5758776/?site_locale=fr_FR`.

9 Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity*. Monographs in Computer Science. Springer, 1999. `doi:10.1007/978-1-4612-0515-9`.

10 Heinz-Dieter Ebbinghaus, Jörg Flum, and Wolfgang Thomas. *Mathematical logic (2. ed.)*. Springer, 1994.

11 Thomas Eiter, Yuri Gurevich, and Georg Gottlob. Existential second-order logic over strings. *J. ACM*, 47(1):77–131, 2000. `doi:10.1145/331605.331609`.

12 Ronald Fagin. Generalized first-order spectra and polynomial-time recognizable sets. *Complexity of Computation*, 7:43–74, 1974.

13 Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Springer, 2006. `doi:10.1007/3-540-29953-X`.

14 Fedor V. Fomin, Petr A. Golovach, and Dimitrios M. Thilikos. On the parameterized complexity of graph modification to first-order logic properties. *Theory Comput. Syst.*, 64(2):251–271, 2020. `doi:10.1007/s00224-019-09938-8`.

15 Georg Gottlob, Phokion G. Kolaitis, and Thomas Schwentick. Existential Second-Order Logic Over Graphs: Charting the Tractability Frontier. *Journal of the ACM*, 51(2):312–362, 2004. `doi:10.1145/972639.972646`.

16 Christos H. Papadimitriou. *Computational complexity*. Addison-Wesley, 1994.

17 Till Tantau. Existential second-order logic over graphs: A complete complexity-theoretic classification. In *32nd International Symposium on Theoretical Aspects of Computer Science, STACS 2015, March 4-7, 2015, Garching, Germany*, pages 703–715, 2015. `doi:10.4230/LIPIcs.STACS.2015.703`.