Dichotomies and Duality in First-order Model Checking Problems

Barnaby Martin
Department of Computer Science, University of Durham,
Science Labs, South Road, Durham DH1 3LE, U.K.
b.d.martin@durham.ac.uk

Abstract

We study the complexity of the model checking problem, for fixed model A, over certain fragments $\mathcal{L}$ of first-order logic. These are sometimes known as the expression complexities of $\mathcal{L}$. We obtain various complexity classification theorems for these logics $\mathcal{L}$ as each ranges over models A, in the spirit of the dichotomy conjecture for the Constraint Satisfaction Problem – which itself may be seen as the model checking problem for existential conjunctive positive first-order logic.

1 Introduction

The *model checking problem* over a logic $\mathcal{L}$ takes as input a structure A and a sentence φ of $\mathcal{L}$, and asks whether $A \models \varphi$. The problem can also be parameterised, either by the sentence φ, in which case the input is simply A, or by the model A, in which case the input is simply φ. Vardi has studied the complexity of this problem, principly for logics which subsume **FO**, in [9]. He describes the complexity of the unrestricted problem as the *combined complexity*, and the complexity of the parameterisation by the sentence (respectively, model) as the *data complexity* (respectively, *expression complexity*). For the majority of his logics, the expression and combined complexities are comparable, and are one exponential higher than the data complexity. In this paper, we will be interested in taking certain fragments $\mathcal{L}$ of **FO**, and studying the complexities of the parameterisation of the model checking problem by the model A, that is the expression complexities for certain A.

When $\mathcal{L}$ is the *positive existential conjunctive* fragment of **FO**, $\{\wedge, \exists\}$-**FO**, the model checking problem is equivalent to the much-studied *constaint satisfaction problem* (CSP). The parameterisation of this problem by the model A is equivalent to what is sometimes described as the *non-uniform* constraint satisfaction problem, $\mathrm{CSP}(A)$ [6]. It has been conjectured [4, 2] that the class of CSPs exhibits *dichotomy* – that is, $\mathrm{CSP}(A)$ is always either in P or is NP-complete, depending on the model A. This is tantamount to the condition that the expression complexity for $\{\wedge, \exists\}$-**FO** on A is always either in P or is NP-complete. While in general this conjecture remains open, it has been proved for certain classes of model A. Of particular interest to us is Hell and Nešetřil's dichotomy for undirected graphs: in [5] it is proved that $\mathrm{CSP}(A)$ is in P, if A has a self-loop or is bipartite, and is NP-complete, if A is any other undirected graph.

Owing to the natural duality between $\wedge, \exists$ and $\vee, \forall$, we consider also various dual fragments. For example, the dual of $\{\wedge, \exists\}$-**FO** is *positive universal disjunctive* **FO**, $\{\vee, \forall\}$-**FO**. It is straightforward to see that this class of expression complexities exhibits dichotomy between P and $\mathsf{co\text{-}NP}$-complete if, and only if, the class of CSPs exhibits dichotomy between P and NP-complete.

This paper is organised as follows. Section 2 is devoted to preliminaries and Section 3 to fragments of **FO** whose model checking problems are of low complexity. In Section 4, we consider those fragments that are related to CSPs and their duals. In the case of positive existential conjunctive **FO**, it makes little difference whether or not equality is allowed, that is the expression complexities for $\{\wedge, \exists\}$-**FO** and $\{\wedge, \exists, =\}$-**FO** are equivalent. The same is not true of positive universal conjunctive **FO**: while a classification of the expression complexities over $\{\vee, \forall\}$-**FO** is equivalent to the unproven CSP dichotomy conjecture, we are able to give a full dichotomy for the expression complexities over $\{\vee, \forall, =\}$-**FO**. The reason for this is that the equality relation in the latter somehow simulates a disequality relation in the former. In Section 5, we consider fragments with a single quantifier, but both conjunction and disjunction: in all cases we are able to give dichotomies for the respective classes of expression complexities. Finally, in Section 6, we consider the scope for further work.

2 Preliminaries

In this paper, we consider only finite, non-empty relational structures. A signature σ is a finite sequence of relation symbols $R_1, \ldots, R_j$, with respective arities $a_1, \ldots, a_j$. A σ-structure A consists of a finite, non-empty set $|A|$ – the universe or domain of A – together with some sets $R_1^A \subseteq |A|^{a_1}, \ldots, R_j^A \subseteq |A|^{a_j}$. When the model A is clear, we may drop the superscript, blurring the distinction between the relation *actual* and the relation *symbol*. We denote the cardinality of the domain of A by $||A||$. We generally use x, y, z to refer to elements of A, and $v_1, v_2, \ldots$ to refer to variables that range over those elements. A *digraph* is any structure over the signature that contains a single, binary relation E. An *undirected graph* is a digraph whose edge relation is symmetric.
Let R^A be a k-ary relation on A. We say that R^A is *x-valid*, for some $x \in A$, if $x^k := (x, \ldots, x) \in R^A$. If all k-tuples $\mathbf{t} \in R^A$ are such that $\mathbf{t}$ contains k distinct elements of A, then we say R^A is *antireflexive*. On digraphs, the edge relation is antireflexive if, and only if, it is not x-valid for any x.
For a σ-structure A, we define its complement $\overline{A}$ as that structure having the same universe as A, but whose relations are the (set-theoretic) complements of the relations of A. That is, for each R_i^A, the relation $R_i^{\overline{A}}$ is defined by $\mathbf{x} \in R_i^{\overline{A}}$ iff $\mathbf{x} \notin R_i^A$.
For a σ-structure A, consider the indices i for which R_i^A is non-empty. Let these be, in ascending order, $\alpha_1, \ldots, \alpha_k$. Define the *canonical relation* R_A (note the subscript), of arity $a_{\alpha_1} + \ldots + a_{\alpha_k}$, to be

$$\begin{array}{rl} R^A_{\alpha_1}(v_1, \ldots, v_{a_{\alpha_1}}) & \wedge\ R^A_{\alpha_2}(v_{a_{\alpha_1}+1}, \ldots, v_{a_{\alpha_1}+a_{\alpha_2}})\ \wedge\ \ldots \\ \ldots & \wedge\ R^A_{\alpha_k}(v_{a_{\alpha_1}+\ldots+a_{\alpha_{k-1}}+1}, \ldots, v_{a_{\alpha_1}+\ldots+a_{\alpha_k}}) \end{array}$$

If, and only if, all relations of A are empty, then we set R_A to be $\emptyset$.
If A and B are σ-structures, then a *homomorphism* from A to B is some function $h : |A| \to |B|$ s.t. for all relations R_i and all $(t_1, \ldots, t_{a_i}) \in |A|^{a_i}$, if $(t_1, \ldots, t_{a_i}) \in R_i^A$ then $(h(t_1), \ldots, h(t_{a_i})) \in R_i^B$. We denote the existence of a homomorphism from A to B by $A \to B$. If we have both $A \to B$ and $B \to A$ then A and B are said to be *homomorphically equivalent*. A *retraction* of a structure A is a homomorphism from A to some induced substructure $B \subseteq A$: and if such exists, B is said to be a *retract* of A. The *core* of a structure A is a minimal (w.r.t. size) retract of A (we talk of *the* core since it is known that this is unique up to isomorphism). Let K_n be the complete antireflexive digraph (i.e. clique) on n vertices. We call an undirected graph *bipartite* if its core is either K_2 or K_1.
We consider first-order logic **FO** to be built over the alphabet $\Gamma_1 \cup \Gamma_0$, where $\Gamma_1 := \{\neg, \wedge, \vee, \exists, \forall, =\}$ and $\Gamma_0 := \{(,), R, v, 0, 1\}$, in an inductive manner. For each R_i in σ, and any natural numbers $j_1, \ldots, j_{a_i}$, $R_i(v_{j_1}, \ldots, v_{j_{a_i}})$ is a *formula* with *free variables* $v_{j_1}, \ldots, v_{j_{a_i}}$ (where each relation R_i and variable v_j is coded as $R\,bin(i)$ and $v\,bin(j)$, where $bin(i)$ and $bin(j)$ are the binary representations of i and j, respectively). Likewise, for any j_1, j_2, $v_{j_1} = v_{j_2}$ is a formula with free variables v_{j_1}, v_{j_2}. If φ and ψ are formulae, then $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$ and $(\neg\varphi)$ are also formulae, in each case having as free variables exactly those variables free in the constituent components. Finally, if the formula φ contains the free variable v_j, then $(\exists v_j \varphi)$ and $(\forall v_j \varphi)$ are formulae, whose free variables are exactly those of φ less v_j. A *sentence* is a formula with no free variables.

We will be interested in fragments of **FO** that derive from restricting which of the symbols of $\Gamma_1 := \{\neg, \wedge, \vee, \exists, \forall, =\}$ we permit. In this paper we will concern ourselves with the non-trivial positive fragments involving exactly one quantifier. For $\Gamma \subseteq \Gamma_1$, we denote by Γ-**FO** that fragment of **FO** that is restricted to the symbols of $\Gamma \cup \Gamma_0$. We have 12 cases to consider.

Class I	Class II	Class III
$\{\vee, \exists\}$	$\{\wedge, \exists\}$	$\{\wedge, \vee, \exists\}$
$\{\vee, \exists, =\}$	$\{\wedge, \exists, =\}$	$\{\wedge, \vee, \exists, =\}$
$\{\wedge, \forall\}$	$\{\vee, \forall\}$	$\{\wedge, \vee, \forall\}$
$\{\wedge, \forall, =\}$	$\{\vee, \forall, =\}$	$\{\wedge, \vee, \forall, =\}$

For some $\Gamma \subseteq \Gamma_1$, and for some structure A, we define the *model checking problem* Γ-MC(A) to have as input a sentence φ of Γ-**FO**, and as yes-instances those sentences such that $A \models \varphi$. The complexity of the model checking problem Γ-MC(A) may be termed the expression complexity for Γ-**FO** on A, in line with the parlance of [9]. The following is basic and may easily be verified.

Lemma 1. *For each $\Gamma \subseteq \Gamma_1$, the recognition problem for well-formed sentences of Γ-**FO** is in* Logspace. *Moreover, given any sentence $\varphi \in \Gamma$-**FO**, we may compute in logarithmic space an equivalent sentence φ' in prenex normal form.*

In this paper, we will not be concerned with complexities beneath Logspace. In light of the previous lemma, and w.l.o.g., we henceforth assume all inputs are in prenex normal form.
The following characterisation is insinuated in [9], and, together with the dichotomy conjecture for CSP, provides much of the motivation for the present work.

Proposition 2. *In full generality, the class of problems $\{\neg, \wedge, \vee, \exists, \forall, =\}$-MC($A$), i.e. Γ_1-MC(A), exhibits dichotomy: if $||A|| = 1$ then the problem is in* Logspace*, otherwise it is* Pspace*-complete.*

3 Logics of Class I

We commence with the low-complexity logics of Class I. Let us consider the problem $\{\vee, \exists\}$-MC(G), for some digraph G of size n. An input for this problem will be of the form:

$$\varphi \ := \ \exists \mathbf{v} \ E(v_1, v_1') \vee \ldots \vee E(v_m, v_m')$$

where $v_1, v_1', \ldots, v_m, v_m'$ are the not necessarily distinct variables that comprise $\mathbf{v}$. Now, $G \models \varphi$ iff it contains an edge. The example demonstrates the triviality of the model checking problem on the fragment $\{\vee, \exists\}$-**FO**; the following proves it.

Proposition 3. *Let Γ-**FO** be any of the logics of Class I. For all structures A, the model checking problem Γ-MC(A) is in* Logspace.

Proof. For $\{\vee, \exists\}$-MC(A) and $\{\vee, \exists, =\}$-MC(A): let a be the maximum arity of the relations of A. Consider any prenex sentence for the model checking problem. Suppose $||A|| = n$: we may cycle through each of the n^a tuples in $|A|^a$ looking for a tuple that satisfies some disjunct (for relations of arity less than a, we consider prefix sub-tuples). If we find no such a-tuple, the input is a no-instance, otherwise it is a yes-instance. This requires space $a \log n$, and the result follows.
For $\{\wedge, \forall\}$-MC(A) and $\{\wedge, \forall, =\}$-MC(A), the proof is similar, except that we search for a tuple which falsifies some conjunct: if we find no such tuple, the input is a yes-instance, otherwise it is a no-instance. □

4 Logics of Class II

4.1 $\{\wedge, \exists\}$-FO and $\{\wedge, \exists, =\}$-FO

Owing to the rule of substitution, the logics $\{\wedge, \exists\}$-**FO** and $\{\wedge, \exists, =\}$-**FO** are very nearly identical. We have the trivial inclusion $\{\wedge, \exists\}$-**FO** $\subseteq$ $\{\wedge, \exists, =\}$-**FO**. For the converse, consider any sentence of $\{\wedge, \exists, =\}$-**FO** that contains an extensional relation *that is not equality*. We may remove each instance of an equality $v_i = v_j$ and substitute all instances of v_j with v_i elsewhere in the sentence. Plainly, this sentence is equivalent to the original and is in $\{\wedge, \exists\}$-**FO**. We are left with the degenerate case of a sentence of $\{\wedge, \exists, =\}$-**FO** whose only relations are equalities. Such a sentence will be logically equivalent to $\exists v_1\ v_1 = v_1$, which is true on all models. It should be clear to the reader that the only structures on which $\{\wedge, \exists\}$-**FO** and $\{\wedge, \exists, =\}$-**FO** are not equivalent are those in which all relations are empty. It follows that for structures A in which all relations are empty, while the problem $\{\wedge, \exists\}$-MC(A) is genuinely trivial (has no yes-instances), the problem $\{\wedge, \exists, =\}$-MC(A) is only very nearly trivial (sentences which contain only equalities form exactly the yes-instances). For the purposes of complexity analysis, we consider these logics equivalent.

The model checking problem $\{\wedge, \exists\}$-MC(A) for inputs in prenex form is exactly the non-uniform constraint satisfaction problem CSP(A) (e.g. see [1]). We note that this problem is always in NP: we may guess a satisfying assignment and verify in polynomial time. As we have mentioned, there is a conjectured dichotomy for CSP(A), namely that each instance is either in P or is NP-complete [4, 2]. This remains unproved. The following is a straightforward consequence of our definitions.

Proposition 4. *The class of problems $\{\wedge, \exists\}$-MC(A) exhibits dichotomy between those cases that are in* P *and those that are* NP*-complete, if, and only if, the class of non-uniform constraint satisfaction problems* CSP(A) *exhibits the same dichotomy.*

4.2 $\{\vee, \forall\}$-FO

This logic is dual to the logic $\{\wedge, \exists\}$-**FO** in the following sense. Consider a prenex sentence φ of $\{\vee, \forall\}$-**FO**, where the variables among $\mathbf{v_1}, \ldots, \mathbf{v_m}$ are exacly those of $\mathbf{v}$:

$$\varphi \ := \ \forall \mathbf{v}\ R_{\alpha_1}(\mathbf{v_1}) \vee \ldots \vee R_{\alpha_m}(\mathbf{v_m})$$

Now, $A \not\models \varphi$ iff

$$\begin{array}{lllll} A & \not\models & & \forall \mathbf{v} & R_{\alpha_1}(\mathbf{v_1}) \vee \ldots \vee R_{\alpha_m}(\mathbf{v_m}) & \text{iff} \\ A & \not\models & \neg & \exists \mathbf{v} & \neg[R_{\alpha_1}(\mathbf{v_1}) \vee \ldots \vee R_{\alpha_m}(\mathbf{v_m})] & \text{iff} \\ A & \not\models & \neg & \exists \mathbf{v} & \neg R_{\alpha_1}(\mathbf{v_1}) \wedge \ldots \wedge \neg R_{\alpha_m}(\mathbf{v_m}) & \text{iff} \\ A & \models & & \exists \mathbf{v} & \neg R_{\alpha_1}(\mathbf{v_1}) \wedge \ldots \wedge \neg R_{\alpha_m}(\mathbf{v_m}) & \text{iff} \\ \overline{A} & \models & & \exists \mathbf{v} & R_{\alpha_1}(\mathbf{v_1}) \wedge \ldots \wedge R_{\alpha_m}(\mathbf{v_m}) & \text{iff} \end{array}$$

$\overline{A} \models \varphi'$, where

$$\varphi' \ := \ \exists \mathbf{v}\ R_{\alpha_1}(\mathbf{v_1}) \wedge \ldots \wedge R_{\alpha_m}(\mathbf{v_m}).$$

We can see that the problems $\{\vee, \forall\}$-MC(A) and $\{\wedge, \exists\}$-MC($\overline{A}$) are intimately related. Indeed, the complement of the problem $\{\vee, \forall\}$-MC(A) is equivalent to the problem $\{\wedge, \exists\}$-MC($\overline{A}$) under the reduction which swaps $\forall$ for $\exists$ and $\vee$ for $\wedge$. This reduction is extremely basic (certainly in Logspace) and demonstrates that $\{\vee, \forall\}$-MC(A) is always in co-NP. We also see that, if we choose some A such that $\{\wedge, \exists\}$-MC($\overline{A}$) is NP-complete, then $\{\vee, \forall\}$-MC(A) is co-NP-complete. The following is now elementary.

Proposition 5. *The class of problems $\{\vee, \forall\}$-MC(A) exhibits dichotomy [between* P *and* co-NP*-complete] if, and only if, the class of non-uniform constraint satisfaction problems* CSP(A) *exhibit dichotomy [between* P *and* NP*-complete].*

4.3 $\{\vee, \forall, =\}$-**FO**

This logic is not dual to the logic $\{\wedge, \exists, =\}$-**FO** in the sense just described. Rather it is dual to the logic $\{\wedge, \exists\}$-**FO** when augmented with a *disequality* relation. Since a disequality relation on a structure is tantamount to a graph clique, we are immediately lead to the following.

Proposition 6. *For structures A such that $||A|| \geq 3$, the problem $\{\vee, \forall, =\}$-MC(A) is co-NP-complete.*

Proof. For any A, membership of co-NP follows as in the previous section. Now let $||A|| = n \geq 3$. We will prove that $\{\vee, \forall, =\}$-MC(A) is co-NP-complete by reduction from the complement of the NP-complete *graph n-colourability* problem [5], $\{\wedge, \exists\}$-MC(K_n). Let an input for $\{\wedge, \exists\}$-MC(K_n) be given, of the form:

$$\varphi' \ := \ \exists \mathbf{v} \ E(v_1, v_1') \wedge \ldots \wedge E(v_m, v_m')$$

where $v_1, v_1', \ldots, v_m, v_m'$ are the not necessarily distinct variables that comprise $\mathbf{v}$. Now, in a similar vein to the previous section, $K_n \not\models \varphi'$ iff

$$\begin{array}{llll} K_k & \not\models & \exists \mathbf{v} & E(v_1, v_1') \wedge \ldots \wedge E(v_m, v_m') \quad \text{iff} \\ \overline{K_k} & \models & \forall \mathbf{v} & E(v_1, v_1') \vee \ldots \vee E(v_m, v_m') \quad \text{iff} \\ A & \models & \forall \mathbf{v} & v_1 = v_1' \vee \ldots \vee v_m = v_m' \end{array}$$

which may be given as an input for the problem $\{\vee, \forall, =\}$-MC(A). □

Thanks to Schaefer [8] we can go further. For the further definitions required for the following, see the appendix.

Theorem 7. *In full generality, the class of problems $\{\vee, \forall, =\}$-MC(A) exhibits dichotomy, between those cases that are in P and those that are co-NP-complete. Specifically:*

- *If $||A|| = 1$, then the problem $\{\vee, \forall, =\}$-MC(A) is in P.*
- *If $||A|| = 2$ then*

  if $R_{\overline{A}}$ is 0-valid, 1-valid, horn, dual horn, bijunctive or affine, then $\{\vee, \forall, =\}$-MC(A) is in P, otherwise

  $\{\vee, \forall, =\}$-MC(A) is co-NP-complete.
- *If $||A|| \geq 3$, then the problem $\{\vee, \forall, =\}$-MC(A) is co-NP-complete.*

Proof. For $||A|| = 1$ each relation R_i^A is either empty or contains the single tuple x^{a_i}, where x is the sole elements of A. A sentence $\varphi :=$

$$\forall \mathbf{v} \ R_{\alpha_1}(\mathbf{v_1}) \vee \ldots \vee R_{\alpha_m}(\mathbf{v_m})$$

(where the variables among $\mathbf{v_1}, \ldots, \mathbf{v_m}$ are exacly those of $\mathbf{v}$ and, say, R_0 is considered to be equality) may readily be evaluated on A by forgetting the quantifiers and substituting for empty relations boolean false (0) and for non-empty relations boolean true (1). This leaves a boolean disjunction that is true iff it contains a disjunct 1, i.e. iff φ contained at least one non-empty relation. This is certainly achievable in Logspace.

The case $||A|| \geq 3$ follows from the previous proposition. The case $||A|| = 2$ follows from our duality together with Schaefer's dichotomy theorem for generalised satisfiability [8]. □

5 Logics of Class III

5.1 $\{\wedge, \vee, \exists\}$-FO

The logics $\{\wedge, \vee, \exists\}$-**FO** and $\{\wedge, \vee, \exists, =\}$-**FO** give rise to model checking problems whose internal structure is not dissimilar to those of $\{\wedge, \exists\}$-**FO** and $\{\wedge, \exists, =\}$-**FO**. For all four logics, the model checking problems are unique up to homomorphism equivalence.

Lemma 8. *The following are equivalent.*

- (i) *The structures A and B are homomorphically equivalent.*
- (ii) *The structures A and B have isomorphic cores.*
- (iii) *The problems $\{\wedge, \exists\}$-$\mathrm{MC}(A)$ and $\{\wedge, \exists\}$-$\mathrm{MC}(B)$ coincide.*
- (iv) *The problems $\{\wedge, \exists, =\}$-$\mathrm{MC}(A)$ and $\{\wedge, \exists, =\}$-$\mathrm{MC}(B)$ coincide.*
- (v) *The problems $\{\wedge, \vee, \exists\}$-$\mathrm{MC}(A)$ and $\{\wedge, \vee, \exists\}$-$\mathrm{MC}(B)$ coincide.*
- (vi) *The problems $\{\wedge, \vee, \exists, =\}$-$\mathrm{MC}(A)$ and $\{\wedge, \vee, \exists, =\}$-$\mathrm{MC}(B)$ coincide.*

Proof. The equivalence of (i) and (ii) follows from the definitions. Each of the impications $(vi), (v), (iv) \rightarrow (i)$ follows from the well-documented $(iii) \rightarrow (i)$ [4].
For the remaining implications, it suffices to prove $(i) \rightarrow (vi)$. We can prove directly that $A \rightarrow B$ implies $\{\wedge, \vee, \exists, =\}$-$\mathrm{MC}(A) \subseteq \{\wedge, \vee, \exists, =\}$-$\mathrm{MC}(B)$ by appealing to the monotonicity of (the quantifier-free part of) $\{\wedge, \vee, \exists, =\}$-**FO**. The same applies with B and A swapped, and the result follows. □

We now turn our attention to $\{\wedge, \vee, \exists\}$-**FO**, returning to $\{\wedge, \vee, \exists, =\}$-**FO** in the next section.

Proposition 9. *Let G be an antireflexive digraph whose edge relation is non-empty. Then $\{\wedge, \vee, \exists\}$-$\mathrm{MC}(G)$ is* NP*-complete.*

Proof. Membership of NP remains elementary; we prove hardness. We may assume w.l.o.g. that G is undirected (symmetric), since otherwise we may define the symmetric closure E' of the edge relation E via: $E'(u, v) := E(u, v) \vee E(v, u)$. More formally, $\{\wedge, \vee, \exists\}$-$\mathrm{MC}(\text{sym-clos}(G))$ easily reduces to $\{\wedge, \vee, \exists\}$-$\mathrm{MC}(G)$ under the reduction which substitutes instances $E^{\text{sym-clos}(G)}(u, v)$ in the former by $E^G(u, v) \vee E^G(v, u)$ in the latter.
Let H be the core of G. Note that the NP-hardness of $\{\wedge, \exists\}$-$\mathrm{MC}(H)$ (a.k.a. $\mathrm{CSP}(H)$) immediately implies the NP-hardness of both $\{\wedge, \exists\}$-$\mathrm{MC}(G)$ and $\{\wedge, \vee, \exists\}$-$\mathrm{MC}(G)$. Since G is antireflexive and undirected, its core H is either K_1 or K_2 or some non-bipartite H'.
The core H can not be K_1, since then the edge relation of G would have been empty.
If the core H is a non-bipartite H', then, by Hell and Nešetřil's theorem [5], the problem $\{\wedge, \exists\}$-$\mathrm{MC}(H')$ is NP-complete, hence NP-hardness of both $\{\wedge, \exists\}$-$\mathrm{MC}(G)$ and
$\{\wedge, \vee, \exists\}$-$\mathrm{MC}(G)$ follows.
It remains for us to consider the case where the core H is K_2. By the previous lemma, it suffices for us to prove that $\{\wedge, \vee, \exists\}$-$\mathrm{MC}(K_2)$ is NP-hard. We define the ternary not-all-equal NAE_3 relation on K_2 in $\{\wedge, \vee, \exists\}$-**FO**, whereupon we may appeal to the NP-hardness of not-all-equal 3-satisfiability (whose inputs may readily be expressed in $\{\wedge, \vee, \exists\}$-**FO**). We give $\mathsf{NAE}_3(u, v, w) := E(u, v) \vee E(v, w) \vee E(w, u)$. □

Proposition 10. *Let A be a structure whose canonical relation R_A is k-ary for some $k \geq 2$. If R_A is antireflexive, then $\{\wedge, \vee, \exists\}$-$\mathrm{MC}(A)$ is* NP*-complete.*

Proof. Note that it follows from the defintion that R_A is non-empty. Consider the binary relation $E(v_1, v_2) := \exists v_3, \ldots, v_k R(v_1, \ldots, v_k)$. This relation specifies a non-empty, antireflexive digraph. The result follows from the previous proposition. $\square$

Proposition 11. *Let A be a structure whose canonical relation R_A is k-ary for some $k \geq 2$. If R_A is not x-valid, for all $x \in A$, then $\{\wedge, \vee, \exists\}$-MC$(G)$ is* NP*-complete.*

Proof. We take R_A and build from it an antireflexive relation R'. Recall that A is fixed, of size $||A|| = n$, and consider its elements to be ordered $x_1, \ldots, x_n$. Take $R^{(0)} := R_A$. From $R^{(m)}$, we build $R^{(m+1)}$ in the following manner. First, we list the tuples of $R^{(m)}$ lexicographically. We proceed through these tuples until we find one that has (at least one instance of) a repeated element. We now build $R^{(m+1)}$ by collapsing all the distinct repeated elements of that tuple to that distinct element. For example, if $R^{(m)}$ is of arity 5, and the first tuple with the desired property is $(x_3, x_2, x_3, x_4, x_4)$, then $R^{(m+1)}(u, v, w) := R^{(m)}(u, v, u, w, w)$. Clearly this process terminates, i.e. a point M is reached where $R^{(M)} = R^{(M+1)}$, and clearly $R^{(M)}$ is antireflexive. Furthermore, by non-x-validity of R_A, for all x, we know that $R^{(M)}$ has arity $k \geq 2$. We set $R' := R^{(M)}$. The result follows from the previous proposition. $\square$

Theorem 12. *The class of problems $\{\wedge, \vee, \exists\}$-MC$(A)$ exhibits dichotomy. Specifically, if R_A is either empty or x-valid, for some $x \in A$, then $\{\wedge, \vee, \exists\}$-MC$(A)$ is in* Logspace*, otherwise it is* NP*-complete.*

Proof. If R_A is empty or x-valid, for some $x \in A$, then it follows that all relations R_i^A are either empty or x-valid. An input for $\{\wedge, \vee, \exists\}$-MC$(A)$ readily translates to an input for the Boolean Sentence Value Problem, under the substitution of 0 and 1 for the empty and x-valid relations, respectively. The Boolean Sentence Value Problem is known to be in Logspace [7].
If R_A is neither empty nor x-valid, for any $x \in A$, then the result follows from the previous proposition. $\square$

5.2 $\{\wedge, \vee, \exists, =\}$-FO

This case is as the previous, via the same proof: although we may note that it is no longer possible for R_A to be empty.

Theorem 13. *The class of problems $\{\wedge, \vee, \exists\}$-MC$(A)$ exhibits dichotomy. Specifically, if R_A is x-valid, for some $x \in A$, then $\{\wedge, \vee, \exists\}$-MC$(A)$ is in* Logspace*, otherwise it is* NP*-complete.*

5.3 $\{\wedge, \vee, \forall\}$-FO

Note that the duality that we introduced in Section 4.2 works, via de Morgan's laws, perfectly well in the presence of both $\wedge$ and $\vee$. The case $\{\wedge, \vee, \forall\}$-**FO** is perfectly dual to $\{\wedge, \vee, \exists\}$-**FO** of the section before last. The following is straightforward.

Theorem 14. *The class of problems $\{\wedge, \vee, \forall\}$-MC$(A)$ exhibits dichotomy. Specifically, if $R_{\overline{A}}$ is either empty or x-valid, for some $x \in A$, then $\{\wedge, \vee, \forall\}$-MC$(A)$ is in* Logspace*, otherwise it is* co-NP*-complete.*

5.4 $\{\wedge, \vee, \forall, =\}$-FO

This logic is now dual to the logic $\{\wedge, \vee, \exists\}$-**FO** augmented with a disequality relation.

Theorem 15. *In full generality, the class of problems $\{\wedge, \vee, \forall, =\}$-MC$(A)$ exhibits trichotomy. Specifically:*

- *If $||A|| \geq 3$ then $\{\wedge, \vee, \forall, =\}$-MC$(A)$ is* co-NP*-complete.*
- *If $||A \leq 2$ then: if $R_{\overline{A}}$ is either empty or x-valid, for some $x \in A$, then $\{\wedge, \vee, \forall\}$-MC$(A)$ is in* Logspace*, otherwise it is* co-NP*-complete.*

Proof. The first part follows from Theorem 7 and the second part follows from the previous theorem. $\square$

6 Further Work

In this paper we have not strayed in to those fragments of **FO** which contain both quantifiers. As has been mentioned, the model checking problem for **FO** has been studied [9], and a dichotomy for $\{\neg, \wedge, \vee, \exists, \forall, =\}$-MC$(A)$ obtained: the problem is in Logspace for $||A|| = 1$ and is Pspace-complete otherwise. A classification for $\{\neg, \wedge, \vee, \exists, \forall\}$-MC$(A)$ is also known[1]: the problem is in Logspace if all relations of A are either empty or contain all tuples; and is Pspace-complete otherwise.

Just as $\{\wedge, \exists\}$-MC(A) is exactly the problem CSP(A), $\{\wedge, \exists, \forall\}$-MC$(A)$ is exactly the problem QCSP(A) – the *quantified constraint satisfaction problem* with template A. The classification problem for this class appears to be as difficult as that for the CSP, and while it is known that complexities of P, NP-complete and Pspace-complete are attainable, little is known as to what may be inbetween. We anticipate that, with regard to model checking problems, the fragments $\{\wedge, \exists, \forall, =\}$-**FO**, $\{\vee, \exists, \forall\}$-**FO** and $\{\vee, \exists, \forall, =\}$-**FO** sit in relation to $\{\wedge, \exists, \forall\}$-**FO** as the fragments $\{\wedge, \exists, =\}$-**FO**, $\{\vee, \forall\}$-**FO** and $\{\vee, \forall, =\}$-**FO** sit in relation to $\{\wedge, \exists\}$-**FO**.

This leaves the twin fragments $\{\wedge, \vee, \exists, \forall\}$-**FO** and $\{\wedge, \vee, \exists, \forall, =\}$-**FO**, whose model checking problems may demonstrate the richest variety of complexities. Certainly there are templates A such that $\{\wedge, \vee, \exists, \forall\}$-MC$(A)$ attains each of the complexities P, NP-complete, co-NP-complete and Pspace-complete. However, a full classification resists.

References

[1] Borner, F., Krokhin, A., Bulatov, A., and Jeavons, P. Quantified constraints and surjective polymorphisms. Tech. Rep. PRG-RR-02-11, Oxford University, 2002.

[2] Bulatov, A., Krokhin, A., and Jeavons, P. Constraint satisfaction problems and finite algebras. In *Proceedings 27th International Colloquium on Automata, Languages and Programming, ICALP'00* (2000), vol. 1853 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 272–282.

[3] Chandra, A., and Merlin, P. Optimal implementation of conjunctive queries in relational databases. In 9*th ACM Symposium on Theory of Computing* (1979), pp. 77–90.

[4] Feder, T., and Vardi, M. Y. The computational structure of monotone monadic SNP and constraint satisfaction: a study through datalog and group theory. *SIAM J. Comput. 28* (1999).

[5] Hell, P., and Nešetřil, J. On the complexity of H-coloring. *J. Combin. Theory Ser. B 48* (1990).

[6] Kolaitis, P., and Vardi, M. Conjunctive-query containment and constraint satisfaction. In *Proc. 17th ACM Symp. on Principles of Database Systems* (1998).

[7] Lynch, N. Log space recognition and translation of parenthesis languages. *Journal of the ACM 24* (1977), 583–590.

[8] Schaefer, T. The complexity of satisfiability problems. In *STOC* (1978).

[9] Vardi, M. Complexity of relational query languages. In *14th Symposium on Theory of Computation* (1982).

[1] It is claimed in [9] that a proof of this appears in [3]. We can find no such proof.

7 Appendix: Schaefer's Boolean Relations

A structure or relation is *boolean* if its domain is of size 2. Without loss of generality, we may assume that the elements of the domain are 0 and 1. We may refer to boolean relations by some propositional formula that expresses them, reading the propositional variables lexicographically, e.g. $[A \vee B]$ expresses $\{(0,1),(0,1),(1,1)\}$; $[A \neq B]$ expresses $\{(0,1),(1,0)\}$. A boolean relation R, of arity a, is:

(*i*) 0-*valid* iff it contains the tuple (0^a).

(*ii*) 1-*valid* iff it contains the tuple (1^a).

(*iii*) *horn* iff it may be expressed by a propositional formula in CNF where each clause has at most one positive literal.

(*iv*) *dual horn* iff it may be expressed by a propositional formula in CNF where each clause has at most one negative literal.

(*v*) *bijunctive* iff it may be expressed by a propositional formula in 2-CNF.

(*vi*) *affine* iff it may be expressed by a propositional formula that is the conjunction of linear equations over $\mathbf{Z}_2$.

The following is Schaefer's dischotomy theorem for *generalised satisfiability without constants.*

Theorem 16. *[8] Let A be a boolean structure. Then* $\mathrm{CSP}(A)$ *(equivalently,* $\{\wedge, \exists\}$-$\mathrm{MC}(A)$*) is in* P *if R_A is in any of the six classes above, otherwise it is* NP*-complete.*