Tightening Faster Model Checking of First-Order Graph Properties, via Tarski's Calculus of Relations

YN: Tentative

Yoshiki Nakamura
Science Tokyo, Japan

Yuya Uezato
CyberAgent, Inc., Japan

Abstract

This is a SUPER-STRONG-PAPER and brings a NEW ERA to the field of FO.

2012 ACM Subject Classification Theory of computation → Logic

Keywords and phrases First-order Logic,Fine-grained Complexity

Digital Object Identifier 10.4230/LIPIcs...

1 Introduction

(Background)

- $\mathbb{FO}^k$ モデル検査: in $\mathcal{O}(\|\varphi\| \cdot n^k)$ time (see, *e.g.*, [19, Proposition 3.1])
- Williams' algorithm: in $\mathcal{O}(2^{\|\varphi\|} \cdot n^\omega)$ time for $\mathbb{FO}^3$ [21, Corollary 1.3][9, Theorem 7]. (★: $\mathcal{O}(2^{\|\varphi\|} \cdot n^{k-3+\omega})$ time for $\mathbb{FO}^k$ は正しい？)
- CoR モデル検査: in $\mathcal{O}(\|\varphi\| \cdot n^\omega)$ time.

(Contribution)

- Williams' algorithm (for $\mathbb{FO}^3$ and for $\mathbb{FO}^k$ in PNF) ≈ $\mathbb{FO}^3$-to-CoR translation.
- For $\mathbb{FO}^k$, for any $\varepsilon > 0$, there is no algorithm in $\mathrm{poly}(\|\varphi\|) \cdot n^{k-\varepsilon}$ time, under SETH.
  - There is no polynomial-time translation from $\mathbb{FO}^3$ to CoR, under SETH.
- $\mathrm{poly}(\|\varphi\|) \cdot n^{3-\varepsilon}$ time algorithms for (parameterized) fragments of $\mathbb{FO}^3$.
  - "quantifier-width"
  - A dichotomy *w.r.t.* signatures

...

YN: TODO: (Below is wip.)

The *model checking problem* is the following problem:

given both a sentence φ and a structure $\mathfrak{A}$, does $\mathfrak{A} \models \varphi$ hold?

Unfortunately, the model checking problem for $\mathbb{FO}$ is complete in PSPACE. Nevertheless, when the number of variables is fixed, the model checking problem for $\mathbb{FO}^k$ can be decided in $\mathcal{O}(\|\varphi\| \cdot n^k)$ time by a naive bottom-up evaluation algorithm (see, *e.g.*, [19, Proposition 3.1]), where $\|\varphi\|$ is the length of φ and n is the number of vertices in $\mathfrak{A}$. A natural question is

Can the complexity be improved from $\mathcal{O}(\|\varphi\| \cdot n^k)$ time?

Williams [21] gave a positive answer to this question. He showed that when φ is *fixed*, the following holds:

Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

- For every $k \geq 3$, the model checking problem for $\mathbb{FO}^k$ sentences[1] can be decided in $\mathcal{O}(n^{k-3+\omega})$ time [21, Corollary 1.3][9, Theorem 7], where ω is the exponent of matrix multiplication.
- For some $k \geq 3$,[2] if the model checking problem for $\mathbb{FO}^k$ sentences[3] can be decided in $\mathcal{O}(n^{k-1-\varepsilon})$ time for some $\varepsilon > 0$, then the Strong Exponential Time Hypothesis (SETH) is false [21, Corollary 1.4].

However when φ is not fixed, Williams' algorithm [21] for $\mathbb{FO}^3$ sentences is $\mathcal{O}(2^{\|\varphi\|} \cdot n^{\omega})$ time. The exponential blowup *w.r.t.* $\|\varphi\|$ is due to that the algorithm requires a transformation to the disjunctive normal form (DNF).

...

In this paper, we first revisit Williams' algorithm. We give a perspective from *Tarski's calculus of relations* (henceforth, *CoR*) [17]. ...We can translate $\mathbb{FO}^3$ sentences into CoR sentences in $\mathcal{O}(2^{\|\varphi\|})$ time. The original translation is given in [18]. An $\mathcal{O}(2^{\|\varphi\|})$ time translation is given in [15]; an implementation of this translation is given in [2].[4] For CoR sentences, the model checking problem can be decided in $\mathcal{O}(\|\varphi\| \cdot n^{\omega})$ time by a naive bottom-up evaluation algorithm, where we use the matrix multiplication algorithm for the relational composition (;). Combining them, we have that the model checking problem for $\mathbb{FO}^3$ sentences can be decided in $\mathcal{O}(2^{\|\varphi\|} \cdot n^{\omega})$ time.

Now, ...

YN: TODO: Parameterized complexity (*e.g.*, [7, 6, 5, 8]).
Parameterized complexity for space complexity [3]

2 Preliminaries

We write $\mathbb{Z}$ for the set of all integers.

We write $[l, r]$ for the set $\{i \in \mathbb{Z} \mid l \leq i \leq r\}$.

A *relational signature* (henceforth, signature) σ is ...

A *structure* $\mathfrak{A}$ over a signature σ is a tuple $\langle |\mathfrak{A}|, \{R^{\mathfrak{A}}\}_{R \in \sigma} \rangle$, where its *universe* $|\mathfrak{A}|$ is a finite[5] set of *vertices* and each $R^{\mathfrak{A}}$ is a binary relation on $|\mathfrak{A}|$.

A *sentence* is a formula without free variables.

For a formula φ and a valuation $\mathfrak{v}$, we write $\mathfrak{A}, \mathfrak{v} \models \varphi$ to denote that φ is true at $\mathfrak{A}$ under $\mathfrak{v}$. For a sentence φ, we write $\mathfrak{A} \models \varphi$ to denote that φ is true at $\mathfrak{A}$.

We write $\mathbb{FO}^k$ to denote the set of formulas of First-order predicate logic with k-variables.

---

1 Precisely, Williams [21] shows for $\mathbb{FO}^k$ sentences in *prenex normal form* (so, the number of occurrences of quantifiers is at most k) [21, Corollary 1.3]. Later, Gao and Impagliazzo [9, Theorem 7] claims that Williams' algorithm can be extended for general $\mathbb{FO}^k$ sentences.

YN: (メモ 3)

2 When $k \leq 3$, it is almost clear that the model checking problem cannot be decided in $\mathcal{O}(n^{k-1-\varepsilon})$, as the input adjacent matrix is given by $\Theta(n^2)$ cells. See Proposition 14 for a more precise proof.

3 This claims holds even for $\mathbb{FO}^k$ sentences in $\exists^*\forall$-prenex normal form.

4 Williams' algorithm takes almost the same steps as in the translation in [15]... (TODO: revise the sentences)

5 In this paper, we are only interested in finite structures.

Important Remark (Unary and Binary Predicates)

We allow unary and binary predicates in our $\mathbb{FO}^3$. Therefore, we do not allow terms involving ternary predicates such as $\forall x. \exists y. \forall z. P(x, y, z)$.[6]

On this remark, we can identify each model $\mathcal{M}$ as color graphs as follows:

- Each element corresponds a node (vertex).
- Each unary predicate U_i means "color-i" node.
- Each binary predicate P_j is "color-j" edge.
- For example, on two nodes x, y, a term $(U_i(x) \wedge U_j(y) \wedge P_k(x, y))$ means that x has the color i, y has the color j, and there is a color k edge from x to y.

Quantifier width

Quantifier widthという謎の指標を導入します。
与えられた式の部分項 $\forall/\exists x.(\cdots)$ について $\cdots$ 部分に「直接」出てくるquantifierの個数を、この部分項のquantifier widthと呼ぶことにします。
たとえば、次の部分項

$$\forall x. \left(\left(\underline{\forall} y. P(y) \right) \vee \left(\underline{\forall} z. (\underline{\forall} x. P(z, x)) \wedge Q(z) \right) \right)$$

については、内部に3つのquantifierの出現（下線）があるのですが、直接の出現は赤くした2つだけなので、この部分項のquantifier widthは「２」です。
そして、式のquantifier widthというのは、最大のquantifier widthを持つ部分項における、その数として定義します。

Tarski's Calculus of Relations (CoR)

CoR terms are generated by the following grammar:

$$\begin{aligned} t, s \ ::= {} & a \mid t + s \mid t \cap s \mid t^{-} \\ & \mid t \,;\, s \mid t \dagger s \mid t^{\pi}. \end{aligned}$$

▶ **Proposition 1.** *The model checking problem for atomic CoR formulas is in $\mathcal{O}(\|\varphi\| \cdot n^{\omega})$ time.*

Proof Sketch. By a naive bottom-up evaluation algorithm. Here, we use the matrix multiplication algorithm for the relational composition (;). ◀

SETH

▶ **Conjecture 2** (The *Strong Exponential Time Hypothesis* (*SETH*) [11, 12]). *For every $\delta < 1$, there exists an integer k such that k-CNF-SAT with n variables cannot be solved in $\mathcal{O}(2^{\delta n})$ time.*

SETH implies the following hypothesis.

YN: たとえば [20] では、$\mathcal{O}(2^{\delta n})$ でなく $2^{\delta n}\ \mathrm{poly}(n)$ を用いているが、
予想としては同値（のはず。$\mathrm{poly}(n) < \mathcal{O}(2^{\delta n})$ なので）。
$\mathcal{O}(2^{n}\ \mathrm{poly}(n))$ のアルゴリズムが存在するという fact に合わせている？

YN: However, it is not known whether the two conjectures are equivalent. [4, Previous Work]

---

[6] For each unary predicate U, we can simulate it by a diagonal binary predicate $B_U(x, y) := x = y \wedge U(x)$. However, in our later construction, we use some unary predicates; thus, we also allow unary predicates explicitly.

▶ **Conjecture 3** (*e.g.*, in [16]). *For every $\delta < 1$, CNF-SAT with n variables and m clauses cannot be solved in $2^{\delta n} \cdot \mathrm{poly}(m)$ time.*

3 Faster FO3 model checking and the FO3-to-CoR translation

YN: TODO: Give an outline of the translation [15].

For instance, when φ is the following $\mathbb{FO}^3$ sentence

$$\exists y.((P(x,y) \vee Q(y,z)) \wedge R(x,y)) \wedge S(x,z),$$

we can translate φ into the CoR formula as follows:

$$\begin{aligned}
&\exists y.((P(x,y) \vee Q(y,z)) \wedge R(x,y)) \wedge S(x,z) \\
&\rightsquigarrow \exists y.(P(x,y) \wedge R(x,y) \wedge S(x,z)) \vee (Q(y,z) \wedge R(x,y) \wedge S(x,z)) && \text{(taking the DNF)} \\
&\rightsquigarrow ((\exists y.P(x,y) \wedge R(x,y)) \wedge S(x,z)) \vee ((\exists y.Q(y,z) \wedge R(x,y)) \wedge S(x,z)) \\
& && \text{(pushing quantifiers)} \\
&\rightsquigarrow (((P \cap R) \,;\, \top)(x,z) \wedge S(x,z)) \vee ((R \,;\, Q)(x,z) \wedge S(x,z)) \\
& && \text{(translating into CoR (intermediate))} \\
&\rightsquigarrow ((((P \cap R) \,;\, \top) \cap S) + ((R \,;\, Q) \cap S))(x,z). && \text{(translating into CoR)}
\end{aligned}$$

▶ **Proposition 4** ([15]). *There is an $\mathcal{O}(2^{\|\varphi\|})$-time translation from an $\mathbb{FO}^3$ formula φ with at most two free variables into an atomic CoR formula semantically equivalent to φ.*

Hence, combining with Proposition 1, we have obtained Williams' result for $k = 3$, via the $\mathbb{FO}^3$-to-CoR translation.

▶ **Theorem 5.** *The model checking problem for $\mathbb{FO}^3$ sentences can be decided in $\mathcal{O}(2^{\|\varphi\|} \cdot n^{\omega})$ time.*

Proof. Given an $\mathbb{FO}^3$ sentence φ, we can translate φ into an atomic CoR formula φ' in $\mathcal{O}(2^{\|\varphi\|})$ time (Proposition 4). The size of φ' is also $\mathcal{O}(2^{\|\varphi\|})$. Hence, by Proposition 1, we have obtained an $\mathcal{O}(2^{\|\varphi\|} \cdot n^{\omega})$ time algorithm. ◀

By Proposition 1 and Theorem 5, when the input formula is given by CoR instead of $\mathbb{FO}^3$, the model checking problem is exponentially more efficiently solvable.

4 Complexity gap between FO3 and CoR

A natural question arising from Theorem 5 is whether there is a faster algorithm without the exponential blowup with respect to the length of the input sentence. Namely,

Can we give an $\mathrm{poly}(\|\varphi\|) \cdot n^{3-\varepsilon}$ time algorithm for the model checking problem for $\mathbb{FO}^3$?

In this section, we answer to this question negatively, assuming SETH, as follows.

▶ **Theorem 6.** *Under SETH, for any $k \geq 1$ and any $\varepsilon > 0$,[7] the model checking problem for $\mathbb{FO}^k$ sentences cannot be decided in $\mathrm{poly}(\|\varphi\|) \cdot n^{k-\varepsilon}$ time.*

[7] When $k \leq 2$, the model checking problem cannot be decided in $\mathrm{poly}(\|\varphi\|) \cdot n^{k-\varepsilon}$ time (even when φ is fixed), as the input adjacent matrix is given by $\Theta(n^2)$ cells (Proposition 14, for a more precise proof).

Theorem 6 is a direct consequence of the following lemma.

▶ **Lemma 7.** *Suppose that there are an integer $k \geq 1$, a function f, and an $\varepsilon > 0$ such that the model checking problem for $\mathbb{FO}^k$ sentences is solvable in $\mathcal{O}(f(\|\varphi\|) \cdot n^{k-\varepsilon})$ time. Then CNF-SAT is in $\mathcal{O}(f(\|\varphi\|) \cdot 2^{n(1-\varepsilon/k)})$ time, where $\|\varphi\|$ is the length of the CNF φ and n is the number of variables in φ.*

Proof. Let φ be a CNF with n variables. *W.l.o.g.*, assume that k divides n. Let $p_1, \ldots, p_n$ be variables in φ. We define $\mathfrak{A}$ as the structure with $2^{n/k}$ vertices and unary relation symbols $P_1, \ldots, P_{n/k}$ such that for every distinct pair $\langle v, w \rangle$ of vertices in $\mathfrak{A}$, there is some unary relation symbol P_i such that $v \in P_i^{\mathfrak{A}}$ and $w \notin P_i^{\mathfrak{A}}$ hold or $v \notin P_i^{\mathfrak{A}}$ and $w \in P_i^{\mathfrak{A}}$ hold. We also define ψ as the following $\mathbb{FO}^k$ sentence:

$$\exists z_1. \ldots \exists z_k. \varphi',$$

where φ' is the φ in which each variable p_{kq+r} (where $0 \leq q < n/k$ and $1 \leq r \leq k$) has been replaced with the atomic formula $P_q(z_r)$. For instance, when φ is the following CNF with 6 variables $(p_1 \vee p_3) \wedge (p_5 \vee p_2) \wedge (p_4 \vee p_6)$, the $\mathbb{FO}^3$ sentence ψ is given as follows:

$$\exists z_1. \exists z_2. \exists z_3. (P_1(z_1) \vee P_1(z_3)) \wedge (P_2(z_2) \vee P_1(z_2)) \wedge (P_2(z_1) \vee P_2(z_3)).$$

For each valuation $\mathfrak{v}$ of φ, let $\mathfrak{v}'$ be the (unique) valuation so that for all $0 \leq q < n/k$ and $1 \leq r \leq k$, $\mathfrak{v}(p_{kq+r})$ is true *iff* $\mathfrak{v}'(z_r) \in P_q^{\mathfrak{A}}$. Then, φ is true at $\mathfrak{v}$ *iff* $\mathfrak{A}, \mathfrak{v}' \models \varphi'$. As the map above is a bijection, we have that φ is satisfiable *iff* $\mathfrak{A} \models \psi$. Thus, by applying the assumption, CNF-SAT is in $\mathcal{O}(f(\|\varphi\|) \cdot (2^{n/k})^{k-\varepsilon})$ time. ◀

Another proof of Theorem 6. TODO:

YN: 基本方針は同じだが、unary relation symbols を変数でなく節に対応づけてもよい。

$$\exists z_1. \ldots \exists z_k. \bigwedge_{i=1}^{m} \bigvee_{j=1}^{k} C_i(z_j).$$

（MAX-2SAT to MAX triangle の帰着 [22] を 2^m次元$(0,1)$ベクトルの重みで考えている）
式の長さ：$\mathcal{O}(km)$, モデルの頂点数：$\mathcal{O}(k2^{n/k})$
（Theorem 6 の帰着） 式の長さ：$\mathcal{O}(\|\varphi\|)$, モデルの頂点数：$\mathcal{O}(2^{n/k})$

◀

YN: TODO: Add a note on succinctness.

Moreover, the result above still holds even over the signature with exactly one binary relation symbol.

YN: Theorem 8 uses non-PNF formulas.

▶ **Theorem 8.** *Under SETH, for any $k \geq 1$ and any $\varepsilon > 0$,*[8] *the model checking problem for $\mathbb{FO}^k$ sentences with exactly one binary relation symbol E cannot be decided in poly $(\|\varphi\|) \cdot n^{k-\varepsilon}$ time.*

Theorem 8 is a direct consequence of the following lemma.

[8] When $k \leq 2$, the model checking problem cannot be decided in poly $(\|\varphi\|) \cdot n^{k-\varepsilon}$ time (even when φ is fixed), as the input adjacent matrix is given by $\Theta(n^2)$ cells (Proposition 14, for a more precise proof).

▶ **Lemma 9.** *Suppose that there is an integer $k \geq 2$ and an $\varepsilon > 0$ such that the model checking problem for $\mathbb{FO}^k$ sentences with exactly one binary relation symbol E is solvable in* $\mathrm{poly}(\|\varphi\|) \cdot n^{k-\varepsilon}$ *time. Then CNF-SAT is in* $\mathrm{poly}(\|\varphi\|) \cdot 2^{n(1-\varepsilon/k)}$ *time, where $\|\varphi\|$ is the length of the CNF φ and n is the number of variables in φ.*

Proof. Let φ be a CNF with n variables. *W.l.o.g.*, assume that k divides n. Let $p_1, \dots, p_n$ be variables in φ. We define the structure $\mathfrak{A}$ as follows:

- $|\mathfrak{A}| := (\{\mathrm{V}\} \times [0, 2^{n/k} - 1]) \uplus (\{\mathrm{E}\} \times [1, n/k])$; we write v^{V} for $\langle \mathrm{V}, v\rangle$ (similarly for v^{E}).
- $E^{\mathfrak{A}} := \{\langle v^{\mathrm{V}}, w^{\mathrm{E}}\rangle \mid v \in [0, 2^{n/k} - 1], w \in [1, n/k],$ and the w-th bit of v is $1\} \cup \{\langle w^{\mathrm{E}}, (w+1)^{\mathrm{E}}\rangle \mid w \in [1, n-1]\} \cup \{\langle n^{\mathrm{E}}, n^{\mathrm{E}}\rangle\}$.

We also define ψ as the following $\mathbb{FO}^k$ sentence:

$$\exists z_1 . \dots \exists z_k . \varphi',$$

where φ' is the φ in which each variable p_{kq+r} (where $0 \leq q < n/k$ and $1 \leq r \leq k$) has been replaced with the $\mathbb{FO}^2$ formula $X_q(z_r)$ defined as follows:

$$X_q(z) := \begin{cases} E(z, z) & \text{if } q = 0, \\ \exists z' . E(z, z') \wedge X_{q-1}(z') & \text{if } q > 0 \text{ where } z' \text{ is a variable not equal to } z. \end{cases}$$

Note that $X_q(z_r)$ is semantically equivalent to the CoR formula $(E^q(E \cap \mathsf{I}))^{\mathsf{d}}(z_r, z_r)$.

For instance when $k = 3$, the variable $p_{2\cdot 1+3}$ is transformed to the following formula:

$$\exists z' . E(z_3, z') \wedge (\exists z'' . E(z', z'') \wedge E(z'', z'')).$$

For each valuation $\mathfrak{v}$ of φ, we can take a valuation $\mathfrak{v}'$ of φ' so that for all $0 \leq q < n/k$ and $1 \leq r \leq k$, $\mathfrak{v}(p_{kq+r})$ is true *iff* $\mathfrak{A}, \mathfrak{v}' \models X_q(z_r)$. Then, φ is true at $\mathfrak{v}$ *iff* $\mathfrak{A}, \mathfrak{v}' \models \varphi'$. Conversely, for each valuation $\mathfrak{v}'$ of φ', we can take a valuation $\mathfrak{v}$ of φ so that φ is true at $\mathfrak{v}$ *iff* $\mathfrak{A}, \mathfrak{v}' \models \varphi'$. Thus, we have that φ is satisfiable *iff* $\mathfrak{A} \models \psi$. Thus, by applying the assumption, CNF-SAT is in $\mathrm{poly}(n/k \cdot \|\varphi\|) \cdot (2^{n/k} + n/k)^{k-\varepsilon}$ time.As $n \leq \|\varphi\|$, this implies $\mathrm{poly}(\|\varphi\|) \cdot 2^{n(1-\varepsilon/k)}$. ◀

YN: TODO: Add a dichotomy (メモ 3) in some section.

5 FPT on FO3

We write $\mathbb{FO}^3_{k,w}$ to denote the set of formulas of $\mathbb{FO}^3$ with k-predicates and w-quantifier-width.

▶ **Theorem 10** ($\mathbb{FO}^3_{k,w}$ **model checkig is FPT).**

Let $\varphi : \mathbb{FO}^3_{k,w}$.

For a given model (graph) $\mathcal{G}$, we can decide if $\mathcal{G} \models \varphi$ in $O(2^{k+w} \cdot |\varphi| \cdot N^{\omega})$ where

- *N is the number of vertices in $\mathcal{G}$; and*
- *ω is a complexity constant for the boolean matrix multiplication (BMM): i.e., $O(N^{\omega})$-time is the time complexity for BMM of two (N, N) matrices.*[9]

▶ **Corollary 11.** *If we fix the number of predicates k and width w, the model checking problem $\mathcal{G} \models \varphi$ in $O(|\varphi| \cdot N^{\omega})$ time-complexity where N is the number of nodes in $\mathcal{G}$.*

[9] 何らかの論文をciteして、今の記録がどれくらいか書いておくと分かりやすい

> kをfixしない、素朴な $\{(x,y,z)\}$ 構築アルゴリズムだと $O(|\varphi| \cdot N^3)$ になって、 その設定だと Theorem 10 は 係数 が大きくなりすぎて遅いということを、 意味のある形で言及した方が良い。

以降、Theorem 10 の証明をやっていきます。

5.1 Proof of Theorem 10

Here we use $P, Q, R \ldots$ to denote binary predicates and $U, \ldots$ for unary predicates.

For a given graph (model) $\mathcal{G}$, we write $N_{\mathcal{G}}$ or simply N to denote the number of vertices of $\mathcal{G}$.

Our idea: Quantifier elimination (Simple case)

Our query evaluation strategy is evaluating the innermost (quantifier-free formula) to the outermost along with quantifier eliminating.

Let us consider the following situation:

$$Qx.\cdots(Qy.\cdots(Qz.\cdots(Qx.\cdots(Qy.\cdots(\exists x.(P(y,x) \wedge R(x,z)))))))$$

To eliminate $\exists z.P(y,x) \wedge R(x,z)$, we introduce a new predicate defined as follows:

$$S(x,y) := \exists z.P(x,z) \wedge Q(z,y).$$

It is worth noting that S is just the composed predicate of P and Q. So, we write $P \odot Q$ instead of S.

Using this predicate, we can rewrite the above one to the following one:

$$Qx\cdots(Qy\cdots(Qz\cdots(Qx.\ (Qy.\ (P \odot R)(y,z)))))$$

Continuing this process, if we eventually enter $\forall x.\exists y.T(x,y)$:

1. building a unary predicate $T_y(x) := \exists y.T(x,y)$ in $O(N^2)$-time, we rewrite it to $\forall x.T_y(x)$.
2. we then simply evaluate $\forall x.T_y(x)$ as $\bigwedge_{1 \leq i \leq N} T_y(i)$ in $O(N)$-time.

Time-complexity of basic operations

Here we enumerate the time-complexity of our basic operations, which will be used later:

- Building a new predicate $Q(x,y) := P(x,y) \bowtie R(x,y)$ ($\bowtie \in \{\wedge, \vee\}$) needs $O(N^2)$-time.
- Transposing a predicate $P^T(x,y) := P(y,x)$ needs $O(N^2)$-time.
- Evaluating unary predicates $\mathcal{Q}x.P(x)$ ($\mathcal{Q} \in \{\forall, \exists\}$) needs $O(N)$-time.
- Building a new predicate $Q(x) := \mathcal{Q}y.P(x,y)$ ($\mathcal{Q} \in \{\exists, \forall\}$) needs $O(N^2)$-time.

So, the remaining operator is predicate composing $(P \odot Q)(x,y) := \exists z.P(x,z) \wedge Q(z,y)$ from predicates P and Q. Although a very simple matrix multiplication requires $O(N^3)$-time, we can build such $P \odot Q$ using fast matrix multiplication algorithms.

▶ **Proposition 12** (Fact). *From two predicates P and Q, we can build the composited predicate $P \odot Q$ in $O(N^{\omega})$-time.*

5.2 Quantifier elimination: General case

If we encounter terms of the very restricted form like $\exists z.P(x,z) \wedge Q(z,y)$, we can replace it by $(P \odot Q)(x,y)$ with eliminating the quantifier occurrence $\exists z$.

In other words, if we encounter more general form of $\exists z. \cdots$, we need to translate it to some adequate form for quantifier elimination. Let us consider the following term:

$$\exists z.\Big(\big(P(x,z) \vee P(y,z) \vee P(x,y)\big) \wedge \big(Q(x,z) \vee Q(x,y)\big) \wedge \big(R(y,z) \vee R(x,y)\big)\Big).$$

We cannot directly apply our quantifier elimination strategy because $\exists$-quantifiers do not distribute on $\wedge$: i.e., $(\exists z.\varphi_1 \wedge \varphi_2) \neq (\exists z.\varphi_1) \wedge (\exists z.\varphi_2)$.

To apply our quantifier elimination procedure, DNF is an adequate form. However, as well-known, converting-to-DNF translation generates an exponential-large expression in the worst case. For our example, we fist translate the first $(\cdots) \wedge (\cdots)$ subterm as follows:

$$\begin{aligned}
&\big(P(x,z) \vee P(y,z) \vee P(x,y)\big) \wedge \big(Q(x,z) \vee Q(x,y)\big) \\
&\quad \Rightarrow \\
&\quad (P(x,z) \wedge Q(x,z)) \vee (P(y,z) \wedge Q(x,z)) \vee (P(x,y) \wedge Q(x,z)) \\
&\vee (P(x,z) \wedge Q(x,y)) \vee (P(y,z) \wedge Q(x,y)) \vee (P(x,y) \wedge Q(x,y)).
\end{aligned}$$

We continue to normalize this term with $R(y,z) \vee R(x,y)$ and it generates DNF with 12 bases of the form $P(\_,\_) \wedge Q(\_,\_)$.

Since our goal is to design an $O(F(k,w) \cdot |\varphi| \cdot N^{\omega})$-time algorithm, we cannot adopt converting-to-DNF transformations.

5.3 Tseytin-like (?) transformation for General Cases

As we have seen above, explicit translations to DNF does not work well since it may explode given expression to an exponential size.

We here develop a translation inspired by Tseytin's transformation.[10]

実際はインスパイアされたという程でもないんですが、 とはいえよく知られた構成な気もするので、何かciteできるものがあればしたいという感じ。

Let us revisit the above example:

$$\exists z.\Big(\big(P(x,z) \vee P(y,z) \vee P(x,y)\big) \wedge \big(Q(x,z) \vee Q(x,y)\big) \wedge \big(R(y,z) \vee R(x,y)\big)\Big).$$

First, we generate all possible *assignments* to predicates. For our example, we have the following assignments:

	$P(x,z)$	$P(y,z)$	$P(x,y)$	$Q(x,z)$	$Q(x,y)$	$R(y,z)$	$R(x,y)$
α_1	0	0	0	0	0	0	0
α_2	0	0	0	0	0	0	1
α_3	0	0	0	0	0	1	0
$\vdots$							
α	1	0	1	0	1	0	0
$\vdots$							
$\alpha_\star$	1	1	1	1	1	1	1

[10] `https://en.wikipedia.org/wiki/Tseytin_transformation`

Let $\mathcal{A}$ be the set of all assignments.
Let $[\![\Psi]\!]_\alpha$ be a boolean value obtained by evaluating Ψ under an assignment α.
For our example expression ψ, $[\![\psi]\!]_\alpha = 0$ using α appearing in the above table.
If we change α as $\alpha(R(y,z)) \leftarrow 1$, then $[\![\psi]\!]_\alpha = 1$.
Introducing assignment leads to the following useful proposition.

▶ **Proposition 13.**

$$\Psi(x,y,z) \iff \bigvee_{\alpha \in \mathcal{A}} \Big(\alpha \wedge [\![\Psi]\!]_\alpha\Big).$$

Especially, let $\mathcal{V} = \{\alpha \in \mathcal{A} : [\![\Psi]\!]_\alpha\}$, *then we have:*

$$\Psi(x,y,z) \iff \bigvee_{\alpha \in \mathcal{V}} \alpha(x,y,z). \qquad (\star)$$

By the definition of assignments, α takes the following form:

$$\alpha(x,y,z) \equiv P_1(x,y) \wedge P_1(x,z) \wedge P_2(y,z) \wedge \cdots \wedge P_k(x,z).$$

It means that the right-hand side of $(\star)$ takes DNF. Therefore,

$$\exists z.\Psi(x,y,z) \iff (\exists z.\alpha_1(x,y,z)) \vee (\exists z.\alpha_2(x,y,z)) \vee \cdots \vee (\exists z.\alpha_{|\mathcal{V}|}(x,y,z)).$$

It suffices to eliminating an $\exists$-quantifier from $\exists z.\alpha(x,y,z)$ for an assignment α.
We first divide $\alpha(x,y,z)$ into two parts α_z and α_{-z} as follows:

$$\alpha(x,y,z) \equiv \underbrace{(P_1(x,z) \wedge P_2(z,x) \wedge P_1(y,z) \wedge \cdots)}_{\alpha_z : z\text{−involving-terms}} \wedge \underbrace{(P_1(x,y) \wedge P_1(y,x) \wedge P_2 \wedge \cdots)}_{\alpha_{-z} : z\text{−non-involving-terms}}$$

We now have the following:

$$\Big(\exists z.\alpha(x,y,z)\Big) \iff \Big(\alpha_{-z}(x,y) \wedge (\exists z.\alpha_z(x,y,z))\Big).$$

Finally, we transform α_z in the following steps (order):

1. If α_z has a term $P(z,x)$, we replace it with $\widetilde{P}(x,z)$ where $\widetilde{P}(a,b) = P(b,a)$.
2. Similarly, we change terms of the form $P(y,z)$ to $\widetilde{P}(z,y)$.
3. At this point,

$$\alpha_z(x,y,z) \equiv \underbrace{(P_1(x,z) \wedge P_3(x,z) \wedge P_4(x,z) \wedge \cdots)}_{\Psi_x} \wedge \underbrace{(P_2(z,y) \wedge P_3(z,y) \wedge P_5(z,y) \wedge \cdots)}_{\Psi_y}.$$

4. Let Ψ_x (resp. Ψ_y) be the set of predicates appearing in the above former (resp. latter) part.
5. Let us introduce two new predicates:

$$\pi(x,z) := \bigwedge_{P \in \Psi_x} P(x,z), \qquad \lambda(z,y) := \bigwedge_{Q \in \Psi_y} Q(z,y).$$

6. We reach our goal:

$$\alpha_z(x,y,z) = \pi(x,z) \wedge \lambda(z,y).$$

Using the final form, we obtain the following:

$$\begin{aligned}\Big(\exists z.\alpha(x,y,z)\Big) &\iff \Big(\alpha_{-z}(x,y) \wedge (\exists z.\pi(x,z) \wedge \lambda(z,y))\Big) \\ &\iff \Big(\alpha_{-z}(x,y) \wedge (\pi \odot \lambda)(x,y)\Big) \\ &\iff (\alpha_{-z} \wedge (\pi \odot \lambda))(x,y)\end{aligned}$$

5.4 Considering Complexity

To eliminate an $\exists$-quantifier from an assignment α, we introduce a new single predicate $\alpha_{-z} \wedge (\pi \odot \lambda)$.

Now we have a question: To eliminate the $\exists$-quantifier from $\exists z.\Psi(x,y,z)$, from the following characterization

$$\Big(\exists z.\Psi(x,y,z)\Big) \iff \Big(\exists z. \bigvee_{\alpha\in\mathcal{V}} \alpha(x,y,z)\Big) \iff \bigvee_{\alpha\in\mathcal{V}} \Big(\exists z.\alpha(x,y,z)\Big) \iff \bigvee_{\alpha\in\mathcal{V}} (\alpha_{-z}\wedge(\pi\odot\lambda))(x,y)$$

can we avoid to introduce $O(2^k)$ predicates if we have k-predicates?? Please recall that each assignment is a choice from $9k$-literals from $\mathcal{L}$ where

$$\mathcal{L} = \{Q(a,b) : a,b \in \{x,y,z\}, Q \in \{P_1, P_2, \ldots, P_k\}\}.$$

Yes. We can avoid introducing a large number of predicates because we just need to introduce the following single (but large) predicate:

$$(\alpha^1_{-z} \wedge (\pi^1 \odot \lambda^1)) \vee (\alpha^2_{-z} \wedge (\pi^2 \odot \lambda^2)) \vee \cdots \vee (\alpha^n_{-z} \wedge (\pi^n \odot \lambda^n))$$

for $\mathcal{V} = \{\alpha^1, \alpha^2, \ldots, \alpha^n\}$.

Let us consider what happens when repeatedly eliminating quantifiers from the innermost to the outermost:

$$\exists y.\Big((\exists z.\cdots) \bowtie (\exists z.\cdots) \bowtie \cdots \bowtie (\exists z.\cdots)\Big)$$
$$\Rightarrow$$
$$\exists y.\left(\begin{array}{l}\text{Q. how many predicates appear in this scope?}\\ \text{A. } k+w \text{ where } w \text{ is the quantifier width of } \psi\end{array}\right)$$
$$\Rightarrow$$
Eliminating $(\exists y.)$ introduces a single new predicate.

Since the required number of quantifier elimination (= the quantifier depth) is bounded by the size of a given expression $|\psi|$, we obtain the following our theorem in the end.

▶ **Theorem 10** (Restatement). *Let* $\varphi : \mathbb{FO}^3_{k,w}$.
For a given model (graph) $\mathcal{G}$*, we can decide if* $\mathcal{G} \models \varphi$ *in* $O(2^{k+w} \cdot |\varphi| \cdot N^{\omega})$.

6 FO2 model checking

$\mathbb{FO}^2$の論理式 ψ と、グラフ $\mathcal{G}$ について $\mathcal{G} \models \psi$ のモデル検査問題は $O(|\psi| \cdot |\mathcal{G}|)$ で解ける気がします。 ただし、$|\mathcal{G}|$は頂点数ではなく、述語（つまり辺）の定義の記述長も合わせたものです。 面白そうなら書いても良いかと思うのですが、$\mathbb{FO}^3$の話だけにしたほうが、話がぼやけないならなくても良いかなという気がします。
Dense graphについては (x,y) の値の集合を構築する方法で $O(|\psi| \cdot N^2)$ になります。 この N は $\mathcal{G}$ の頂点数の方です。 ただ、sparse graphの時には N^2 は$|\mathcal{G}|$について線形ではないので、そこを改良したいという感じです。
アイディアは、$\mathbb{FO}^3$の時と比べると、$\exists y.P(x,y) \wedge Q(y,x) \simeq (P \odot Q)$ の計算について $(P \odot Q)(x,x)$ の diagonal な形しか要求しないので、行列積を全部やる必要が多分なさそうとかそういう感じです。
Sparse graphの話なので、もしかすると Impagliazzo たちがやっているかもしれません。

YN: [10] Lemma 9.1 Base Case は $O(|\psi| \cdot |\mathcal{G}|)$??

7 Conclusion

References

1. Karl Bringmann, Nick Fischer, and Marvin Künnemann. A fine-grained analogue of schaefer's theorem in P: Dichotomy of $\exists^k\forall$-quantified first-order graph properties. In *CCC*, volume 137 of *LIPIcs*, page 31:1–31:27. Schloss Dagstuhl, 2019. doi:10.4230/LIPIcs.CCC.2019.31.
2. Anthony Brogni and Sebastiaan J. C. Joosten. Translating three-variable first-order predicate logic to relation algebra, implemented using Z3, 2025. doi:10.48550/arXiv.2308.02513.
3. Yijia Chen, Michael Elberfeld, and Moritz Müller. The parameterized space complexity of model-checking bounded variable first-order logic. *Logical Methods in Computer Science*, Volume 15, Issue 3, 2019. doi:10.23638/LMCS-15(3:31)2019.
4. Marek Cygan, Holger Dell, Daniel Lokshtanov, Dániel Marx, Jesper Nederlof, Yoshio Okamoto, Ramamohan Paturi, Saket Saurabh, and Magnus Wahlström. On problems as hard as CNF-SAT. *ACM Trans. Algorithms*, 12(3):41:1–41:24, 2016. doi:10.1145/2925416.
5. Joerg Flum and Martin Grohe. Model-checking problems as a basis for parameterized intractability. *Logical Methods in Computer Science*, Volume 1, Issue 1, 2005. doi:10.2168/LMCS-1(1:2)2005.
6. Jörg Flum and Martin Grohe. Fixed-parameter tractability, definability, and model-checking. *SIAM Journal on Computing*, 31(1):113–145, 2001. doi:10.1137/S0097539799360768.
7. Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2006. doi:10.1007/3-540-29953-X.
8. Markus Frick and Martin Grohe. The complexity of first-order and monadic second-order logic revisited. *Annals of Pure and Applied Logic*, 130(1-3):3–31, 2004. doi:10.1016/J.APAL.2004.01.007.
9. Jiawei Gao and Russell Impagliazzo. The fine-grained complexity of strengthenings of first-order logic, 2019. URL: https://eccc.weizmann.ac.il/report/2019/009/.
10. Jiawei Gao, Russell Impagliazzo, Antonina Kolokolova, and Ryan Williams. Completeness for first-order properties on sparse structures with algorithmic applications. *ACM Trans. Algorithms*, 15(2):23:1–23:35, 2018. doi:10.1145/3196275.
11. Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-SAT. *Journal of Computer and System Sciences*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.
12. Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.
13. François Le Gall, Harumichi Nishimura, and Seiichiro Tani. Quantum algorithms for finding constant-sized sub-hypergraphs. *Theoretical Computer Science*, 609:569–582, 2016. doi:10.1016/j.tcs.2015.10.006.
14. Andrea Lincoln, Virginia Vassilevska Williams, and Ryan Williams. Tight hardness for shortest cycles and paths in sparse graphs. In *SODA*, pages 1236–1252. SIAM, 2018. doi:10.1137/1.9781611975031.80.
15. Yoshiki Nakamura. Expressive power and succinctness of the positive calculus of binary relations. *Journal of Logical and Algebraic Methods in Programming*, 127:100760, 2022. doi:10.1016/j.jlamp.2022.100760.
16. Mihai Pătraşcu and Ryan Williams. On the possibility of faster SAT algorithms. In *SODA*, Proceedings, pages 1065–1075. SIAM, 2010. doi:10.1137/1.9781611973075.86.
17. Alfred Tarski. On the calculus of relations. *The Journal of Symbolic Logic*, 6(3):73–89, 1941. doi:10.2307/2268577.
18. Alfred Tarski and Steven Givant. *A Formalization of Set Theory without Variables*, volume 41. American Mathematical Society, 1987. doi:10.1090/coll/041.
19. Moshe Y. Vardi. On the complexity of bounded-variable queries (extended abstract). In *PODS*, page 266–276. ACM, 1995. doi:10.1145/212433.212474.

20 Virginia Vassilevska Williams. Hardness of easy problems: Basing hardness on popular conjectures such as the strong exponential time hypothesis. In *IPEC*, volume 43 of *LIPIcs*, page 17–29. Schloss Dagstuhl, 2015. doi:10.4230/LIPIcs.IPEC.2015.17.

21 Ryan Williams. Faster decision of first-order graph properties. In *CSL-LICS*, page 1–6. ACM, 2014. doi:10.1145/2603088.2603121.

22 Ryan Williams. Exact algorithms for maximum two-satisfiability. In *Encyclopedia of Algorithms*, pages 683–688. Springer, New York, NY, 2016. doi:10.1007/978-1-4939-2864-4_227.

A Tips

A.1 Trivial Results

▶ **Proposition 14** (Almost trivial fact). *Let φ_0 be the following $\mathbb{FO}^k$ sentence:*

$$\exists x_1 \dots \exists x_k . R(x_1, \dots, x_k).$$

Then for any $\varepsilon > 0$, the model checking problem where the sentence is fixed to φ_0 cannot be decided in $\mathcal{O}(n^{k-\varepsilon})$ time (without any assumptions).

Proof. (Below, we suppose that we can access to at most one cell in each step.) Towards a contradiction, assume that for some M, there is an $Mn^{k-\varepsilon}$ time RAM model $\mathcal{M}$ solving the model checking problem (for sufficiently large n). Then, a sufficiently large $n_0 > n$ satisfies $Mn_0^{k-\varepsilon} < n_0^k$.

Let $\mathfrak{A}$ be the structure with n_0 vertices such that $R^{\mathfrak{A}} = \emptyset$. We consider the run ρ of $\mathcal{M}$ when the input is $\mathfrak{A}$. In the run ρ (of time at most $Mn^{k-\varepsilon}$), as $Mn_0^{k-\varepsilon} < n_0^k$, there is an input cell c for asserting $\langle v_1, \dots, v_k \rangle \in R^{\mathfrak{A}}$ such that ρ does not access c. We then define $\mathfrak{B}$ as the structure $\mathfrak{A}$ in which $R^{\mathfrak{B}}$ has been replaced with the singleton set $\{\langle v_1, \dots, v_k \rangle\}$. When $\mathfrak{B}$ is input, the run is the same as ρ, as the cell c is not accessed in ρ. Hence, in $\mathcal{M}$, the output of $\mathfrak{B}$ is also the same as that of $\mathfrak{A}$, reaching a contradiction. (When $\mathfrak{A}$ is input, the output should be "no". When $\mathfrak{B}$ is input, the output should be "yes".) ◀

YN: 同様の議論がある論文/本からの引用などで簡単に済ませたい。
とくに 3 項関係記号をもつ $\mathbb{FO}^3$ は、$\mathcal{O}(n^{3-\varepsilon})$ で解けない（行列積で解けるのは 2 項関係記号の時）。

▶ **Proposition 15.** *The model checking problem for $\mathbb{FO}^k$ with only unary relation symbols (without equality) is solvable in $\mathcal{O}(n)$ time.*

Proof Sketch. In this case, by taking the DNF, we can transform each subformula of the form $\exists x . \varphi$ into the following form: $\exists x . \bigwedge_i P_i(x)$. Thus, each subformula has at most one free variable. Hence, by the naive evaluation algorithm, we have obtained $\mathcal{O}(n)$ time. ◀

YN: TODO: unary + equality の場合はどうなるか？

A.2 Current Summary

Problem / max. arity k	1	2	3
PNF $\mathbb{FO}^3$ (data)	$\mathcal{O}(n)$ (Proposition 15)	$\mathcal{O}(n^\omega)$ [21]	$\mathcal{O}(n^3)$ (trivial)
$\mathbb{FO}^3$ (data)	no $\mathcal{O}(n^{k-\varepsilon})$ (Proposition 14)		
PNF $\mathbb{FO}^3$ (combined)	$\mathcal{O}(\lVert\varphi\rVert \cdot n^3)$ (trivial) $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n)$ (Proposition 15)	$\mathcal{O}(\lVert\varphi\rVert \cdot n^3)$ (trivial) $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n^\omega)$ [21, 15]	$\mathcal{O}(\lVert\varphi\rVert \cdot n^3)$ (trivial)
$\mathbb{FO}^3$ (combined)	no poly $(\lVert\varphi\rVert) \cdot n^{3-\varepsilon}$ under SETH (Theorem 6)		

Table 1 Complexity for $\mathbb{FO}^3$.

A similar figure can be found in [1, Figure 1] for data complexity (sparse graphs).

B Hyperclique hypothesis and the case of arity $k \geq 4$

/ max. arity k	1	2	3	4
PNF $\mathbb{FO}^4$ (data)	$\mathcal{O}(n)$ (Proposition 15)	$\mathcal{O}(n^{1+\omega})$ [21, Cor. 1.3], $\mathcal{O}(n^{\omega_3})$	$\mathcal{O}(n^{\omega_3})$	$\mathcal{O}(n^4)$ (trivial)
	no $\mathcal{O}(n^{1-\varepsilon})$ (Proposition 14)	no $\mathcal{O}(n^{3-\varepsilon})$ under SETH [21, Cor. 1.4]	no $\mathcal{O}(n^{4-\varepsilon})$ under HyperClique	no $\mathcal{O}(n^{k-\varepsilon})$ (Proposition 14)
$\mathbb{FO}^4$ (data)	$\mathcal{O}(n)$ (Proposition 15)	$\mathcal{O}(n^{\omega_3})$	$\mathcal{O}(n^{\omega_3})$	$\mathcal{O}(n^4)$ (trivial)
	no $\mathcal{O}(n^{1-\varepsilon})$ (Proposition 14)	no $\mathcal{O}(n^{4-\varepsilon})$ under binary HyperClique (Proposition 23) no $\mathcal{O}(n^{3-\varepsilon})$ under SETH [21, Cor. 1.4]	no $\mathcal{O}(n^{4-\varepsilon})$ under HyperClique	no $\mathcal{O}(n^{k-\varepsilon})$ (Proposition 14)
PNF $\mathbb{FO}^4$ (combined)	$\mathcal{O}(\lVert\varphi\rVert \cdot n^4)$ (trivial) $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n)$ (Proposition 15)	$\mathcal{O}(\lVert\varphi\rVert \cdot n^4)$ (trivial) $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n^{1+\omega})$ [21, Cor. 1.3] $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n^{\omega_3})$	$\mathcal{O}(\lVert\varphi\rVert \cdot n^4)$ (trivial) $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n^{\omega_3})$	$\mathcal{O}(\lVert\varphi\rVert \cdot n^4)$ (trivial)
	no poly $(\lVert\varphi\rVert) \cdot n^{4-\varepsilon}$ under SETH (Theorem 6)			
$\mathbb{FO}^4$ (combined)	$\mathcal{O}(\lVert\varphi\rVert \cdot n^4)$ (trivial) $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n)$ (Proposition 15)	$\mathcal{O}(\lVert\varphi\rVert \cdot n^4)$ (trivial) $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n^{\omega_3})$	$\mathcal{O}(\lVert\varphi\rVert \cdot n^4)$ (trivial) $\mathcal{O}(2^{\lVert\varphi\rVert} \cdot n^{\omega_3})$	$\mathcal{O}(\lVert\varphi\rVert \cdot n^4)$ (trivial)
	no poly $(\lVert\varphi\rVert) \cdot n^{4-\varepsilon}$ under SETH (Theorem 6)			

Table 2 Complexity for $\mathbb{FO}^4$.

YN:
- 赤の箇所は未証明だが成り立つと思うもの
- 一般 $\mathbb{FO}^k$ の Williams' algorithm [9, Theorem 7] は怪しい（後述）ので除外

B.1 Boolean matrix multiplication and complexity of the model checking

We recall that

$$(A \cdot B)(x, z) \quad \leftrightarrow \quad \exists y, A(x, y) \wedge B(y, z).$$

Thus, the boolean matrix multiplication can be expressed as a model checking problem for dyadic $\mathbb{FO}^3$. Hence, we can observe the following fact.

▶ **Proposition 16.** *The model checking problem for dyadic $\mathbb{FO}^3$ is solvable in $\mathcal{O}(n^\omega)$ time iff boolean matrix multiplication can be computed in $\mathcal{O}(n^\omega)$ time.*

Proof. $\Longrightarrow$: Matrix multiplication problem can be expressed as a model checking problem. $\Longleftarrow$: By the FO3-to-CoR translation. ◀

Below is a generalization of the boolean matrix multiplication to k-tensors.

▶ **Definition 17** ([14]). *Given k k-tensors of dimensions $n \times \cdots \times n$, $A_1, \ldots, A_k$, the k-wise matrix product is defined as the k-tensor C given by:*

$$C[i_1, \ldots, i_k] \coloneqq \bigvee_{\ell=1}^{n} \bigwedge_{j=1}^{k} A_j[i_1, \ldots i_{j-1}, i_{j+1}, \ldots, i_k, \ell].$$

▶ **Proposition 18.** *The model checking problem for k-adic $\mathbb{FO}^{k+1}$ is solvable in $\mathcal{O}(n^{\omega_k})$ time iff k-wise matrix product can be computed in $\mathcal{O}(n^{\omega_k})$ time.*

Proof. $\Longrightarrow$: k-wise matrix product can be expressed as a model checking problem. $\Longleftarrow$: By an analog of the FO3-to-CoR translation.(TODO:) ◀

By a specific restriction, we can give a matrix multiplication characterization for m-adic $\mathbb{FO}^{k+1}$, as follows:

▶ **Proposition 19.** *The model checking problem for m-adic $\mathbb{FO}^{k+1}$ is solvable in $\mathcal{O}(n^{\omega_{k,m}})$ time iff k-wise matrix product, where the input tensors are $\mathbb{FO}^{k+1}$-defined with m-ary relations, can be computed in $\mathcal{O}(n^{\omega_{k,m}})$ time.*

Proof. $\Longrightarrow$: This restricted k-wise matrix product can be expressed as a model checking problem of m-adic $\mathbb{FO}^{k+1}$. $\Longleftarrow$: Almost trivial. ◀

- Fact: $\omega_{k,k} = \omega_k$。 とくに $\omega_{2,2} = \omega$。
- Fact: $\omega_{k,1} = k$ (成分ごと独立なので)
- Fact: $k = \omega_{k,1} \leq \omega_{k,2} \leq \cdots \leq \omega_{k,k} = \omega_k \leq k+1$
- Fact: $\omega_k = k+1$, under HyperClique Hypothesis [1], for $k \geq 3$

B.2 Hyperclique hypothesis and boolean matrix multiplication

▶ **Hypothesis 20** (($k+1$)-hyperclique hypothesis (rephrased) [14, Hypothesis 1.4])**.** *Let us consider the following formula*

$$\varphi \coloneqq \exists x_1, \dots x_{k+1}, \bigwedge_{j=1}^{k} E(x_1, \dots x_{i-1}, x_{i+1}, \dots, x_{k+1}).$$

Given a structure $\mathfrak{A}$ with one symmetric k-ary relation E (of k-tuples of pairwise distinct elements), does $\mathfrak{A} \models \varphi$ hold? The (k-uniform) ($k+1$)-hyperclique hypothesis is that this problem is not solvable in $\mathcal{O}(n^{k+1-\varepsilon})$ time for any $\varepsilon > 0$. ⌟

For instance, when $k = 3$, the formula in Hypothesis 20 is expressed as follows:

$$\varphi \coloneqq \exists x_1, x_2, x_3,\ E(x_1, x_2, x_3) \wedge \exists x_4, E(x_1, x_2, x_4) \wedge E(x_2, x_3, x_4) \wedge E(x_3, x_1, x_4).$$

To calculate this formula, we can use k-wise matrix product for the part "$\exists x_4, E(x_1, x_2, x_4) \wedge E(x_2, x_3, x_4) \wedge E(x_3, x_1, x_4)$". Hence, we have:

▶ **Proposition 21.** *For $k \geq 3$, under ($k+1$)-hyperclique hypothesis, $\omega_k = k+1$. Hence, the model checking problem for k-adic $\mathbb{FO}^{k+1}$ is not solvable in $\mathcal{O}(n^{k+1-\varepsilon})$ time.*

Below, we introduce a stronger version of the ($k+1$)-hyperclique hypothesis.

▶ **Hypothesis 22** (binary encoded ($k+1$)-hyperclique hypothesis)**.** *For $m \geq 2$, Let us consider the following formula:*

$$\varphi \coloneqq \exists x_1, \dots x_{k+1}, \bigwedge_{j=1}^{k} \mathcal{E}(x_1, \dots x_{i-1}, x_{i+1}, \dots, x_{k+1}),$$

where $\mathcal{E}(y_1, \dots, y_k) \coloneqq \exists z, \bigwedge_{j=1}^{k} E(z, y_j)$. Then, given a structure $\mathfrak{A}$ with one binary relation E, does $\mathfrak{A} \models \varphi$ hold? The binary encoded ($k+1$)-hyperclique hypothesis is that this problem is not solvable in $\mathcal{O}(n^{k+1-\varepsilon})$ time for any $\varepsilon > 0$. ⌟

YN: $\mathcal{E}(y_1, \dots, y_k)$ を2項関係の$\mathbb{FO}^{k+1}$論理式でエンコードする場合を考えています。
愚直に Hypothesis 20 からの帰着を考えると頂点数が n^k になって帰着できない

▶ **Proposition 23.** *For $k \geq 3$, under the binary encoded ($k+1$)-hyperclique hypothesis, $\omega_{k,2} = k+1$. Hence, the model checking problem for dyadic $\mathbb{FO}^{k+1}$ is not solvable in $\mathcal{O}(n^{k+1-\varepsilon})$ time.*

▶ **Problem 24.** *Hypothesis 22 fails?*

YN: $\rightsquigarrow \omega_{k,2}$ の問題の形で書けるが $\mathcal{O}(n^{k+1-\varepsilon})$ が open になっている問題は何かありそうでしょうか？

C memo

1. 2025/06/25 メールより

> あとは、FO3 が $O(2^{\delta\|\varphi\|}n^{\omega})$ time（$\delta < 1$）で解けないことを示せるか？という問題も考えられそうです。（これが示せれば $O(2^{\delta\|\varphi\|})$ time の FO3-to-CoR の変換がないこともいえます）

(Open Problem)

2. 2025/07/04 メールより

YN:

> [21, Theorem 1.3] の大枠の方針は、CoR変換のコアの部分と似た感じで、$\exists y.P(\vec{x}, y) \wedge Q(y, \vec{z})$ の形に持っていって、 *最も内側の* ∃ に対して行列積するというものです。ここで $\vec{x}$ と $\vec{z}$ には以下のような条件を課しておきます。
> - 共通の変数を持たない
> - 次元はおおよそ $(k-1)/2$ ずつに分割されている
>
> この方針を一般に拡張しようとすると、たとえば
>
> $$\exists y.P(x_1, x_2, y) \wedge Q(x_2, x_3, y) \wedge R(x_3, x_2, y)$$
>
> （ただし、各 $P(x_1, x_2, y)$ はたとえば $\exists z.P'(x_1, z) \wedge P'(x_2, z) \wedge P'(z, y)$ のような式）
> みたいな式をどうやって行列積にしますか？という問題が発生しそうです。
> 自由変数が高々2の場合（FO3の場合）には、外側の量化子の自由変数を除くと自由変数が高々1つなので、綺麗な分割ができますが、 そうでなかった場合には、自由変数間の依存が生じるので、行列積に持っていけないのでは？という気がしています。 # 逆に言うとこの辺りの計算量をもう少し厳密に考える価値がありそう？

★ $k \geq 4$ の時に以下は本当に成り立つか？(*cf.* [9, Theorem 7])
- the model checking problem for $\mathbb{FO}^k$ sentences ~~in *prenex normal form*~~ can be decided in $n^{k-1+o(1)}$ time for $k \geq 9$.
- the model checking problem for $\mathbb{FO}^k$ sentences ~~in *prenex normal form*~~ can be decided in $\mathcal{O}(n^{k-3+\omega})$ time for $k \geq 3$.
  - 定数置き換えの方針で、たとえば上の論理式で x_3 を定数としても $P(x_1, x_2, y)$ によって行列積にならない（*cf.* Proposition 14）
  - 下の"関連"の方針から高々$k-1$項関係であれば $\mathcal{O}(n^{\omega_{k-1}})$ time は言えるはず。
    - "k次元行列積"をもつ "k-ary CoR" を考えると同じ議論。
    - したがって、$\omega_{k-1} < k$ の仮説（？）のもと $\mathcal{O}(n^{k-\varepsilon})$ time for some $\varepsilon > 0$.

関連（$\mathbb{FO}^4$ が3項関係をもつ場合、$\mathcal{O}(2^{\|\varphi\|}n^{\omega_3})$ time。 ただし ω_k は以下の"k次元行列積"の時間計算量（$k-1 \leq \omega_k \leq k$）（$\omega_2 = \omega$。 $\omega_k < 1$ を満たすかどうかは $k \geq 3$ で open （？））
- [21, Open Problem 2 (p.5)]

  > "What about for vocabularies with ternary relations? ...We wish to compute the following 3D matrix in $n^{4-\varepsilon}$ time, for some $\varepsilon > 0$:
  >
  > $$D[i, j, k] = \sum_{\ell=1}^{n} A[i, j, \ell] \cdot B[j, k, \ell] \cdot C[k, i, \ell].$$
  >
  > ... would allow us to more efficiently find (for example) a 4-clique in a 3-uniform hypergraph, which can be expressed as a first-order sentence over a ternary relation."
- [13]

  > "... no efficient classical algorithm for 4-clique finding has been discovered so far."
- [14]

  > "the generalized matrix product related to finding hypercliques in k-uniform hypergraphs cannot be sped up with a Strassen like technique"

3. 2025/6/27 メールより

> （補足）結果を組み合わせると、以下の dichotomy を示せそうです（関係記号の個数,定数記号の個数,等号の有無を考えています）
> 【冠頭標準形の FO3 の場合】
> - 一般（＊）： poly $(|\psi|)N^{3-\varepsilon}$ だとSETHに反する ← ② と その修正 （＊）厳密には、1引数関係記号をω個含む場合 もしくは 定数記号をω個含む場合（等号あり） もしくは 2引数関係記号1個以上含むかつ定数記号をω個含む場合
> - 関係記号の個数k固定： $O(|\psi|N^{\omega})$ ← 上里さんの方針
> - 1引数関係記号k個 かつ 定数記号ω個（等号なし）：$\mathcal{O}(|\psi|N^{\omega})$ ← 上里さんの方針（+ 定数記号を先に除去）
>
> 【一般の FO3 の場合】
> - 一般（＊）： poly $(|\psi|)N^{3-\varepsilon}$ だとSETHに反する ← ② と その修正 （＊）厳密には、1引数関係記号をω個含む場合 もしくは 定数記号をω個含む場合（等号あり） もしくは 2引数関係記号1個以上含む場合
> - 1引数関係記号k個 かつ 定数記号ω個（等号なし）：$\mathcal{O}(|\psi|N^{\omega})$ ← 上里さんの方針（+ 定数記号を先に除去 + DNF変換して量化子内側入れの後の論理式 $\exists x \bigwedge_i \alpha_i$ （各 α_i は $P(x)$ or $\neg P(x)$）が有限通り）
> - 1引数関係記号k個 かつ 定数記号k'個（等号あり）：$O(|\psi|N^{\omega})$ ← 上里さんの方針（+ DNF変換して量化子内側入れの後の論理式 $\exists x \bigwedge_i \alpha_i$ （各 α_i は $P(x), \neg P(x), \neg x = y, \neg x = z$, or $\neg x = c$）が有限通り）

⇝ 「定数記号をω個含む場合（等号あり）」は結論自体誤り。[11]
⇝ ひとまず定数記号を考えず、関係記号の個数と等号の有無を考えるなら以下
【冠頭標準形の FO3 の場合】
- 一般（＊）： poly $(|\psi|)N^{3-\varepsilon}$ だとSETHに反する ← ② と その修正 （＊）厳密には、1引数関係記号をω個含む場合（等号なし）
- 関係記号の個数k固定（等号あり）： $O(|\psi|N^{\omega})$ ← 上里さんの方針

【一般の FO3 の場合】
- 一般（＊）： poly $(|\psi|)N^{3-\varepsilon}$ だとSETHに反する ← ② と その修正 （＊）厳密には、1引数関係記号をω個含む場合（等号なし） もしくは 2引数関係記号1個以上含む場合（等号なし）
- 1引数関係記号k個（等号あり）：$O(|\psi|N^{\omega})$ ← 上里さんの方針（+ DNF変換して量化子内側入れの後の論理式 $\exists x \bigwedge_i \alpha_i$ （各 α_i は $P(x), \neg P(x), \neg x = y, \neg x = z$）が有限通り）

[11] この時、各変数の割当は、定数が指す箇所 $\|\psi\|$ 通り + 定数が指さない場合 1 通りのみを考えれば十分なので poly $(\|\psi\|)$ time。（構造のサイズに依存しない）