Ordered fragments of first-order logic

Reijo Jaakkola

University of Helsinki, Tampere University

Funding: Theory of computational logics – Academy of Finland grants 324435 and 328987

The satisfiability problem

- An important invariant of a logic $\mathcal{L}$ is the complexity of its satisfiability problem, i.e., the problem of determining whether a given sentence of $\mathcal{L}$ is satisfiable (in other words has a model).

The satisfiability problem

- An important invariant of a logic $\mathcal{L}$ is the complexity of its satisfiability problem, i.e., the problem of determining whether a given sentence of $\mathcal{L}$ is satisfiable (in other words has a model).
- A classical result of Church and Turing is that the satisfiability problem for full first-order logic FO is undecidable.

The satisfiability problem

- An important invariant of a logic $\mathcal{L}$ is the complexity of its satisfiability problem, i.e., the problem of determining whether a given sentence of $\mathcal{L}$ is satisfiable (in other words has a model).
- A classical result of Church and Turing is that the satisfiability problem for full first-order logic FO is undecidable. This result led researches to focus their study on *fragments* of FO (essentially computable sets $\mathcal{L} \subseteq \mathrm{FO}$) with the hope that expressive logics with decidable satisfiability problem could be identified.

Introduction
Ordered logic
Fluted logic
Syntactical variants
Conclusions

The satisfiability problem

- An important invariant of a logic $\mathcal{L}$ is the complexity of its satisfiability problem, i.e., the problem of determining whether a given sentence of $\mathcal{L}$ is satisfiable (in other words has a model).
- A classical result of Church and Turing is that the satisfiability problem for full first-order logic FO is undecidable. This result led researches to focus their study on *fragments* of FO (essentially computable sets $\mathcal{L} \subseteq \mathrm{FO}$) with the hope that expressive logics with decidable satisfiability problem could be identified.
- An important example of an interesting decidable fragment of FO is the two-variable logic FO^2 (every sentence can contain at most two variables).

The satisfiability problem

- An important invariant of a logic $\mathcal{L}$ is the complexity of its satisfiability problem, i.e., the problem of determining whether a given sentence of $\mathcal{L}$ is satisfiable (in other words has a model).
- A classical result of Church and Turing is that the satisfiability problem for full first-order logic FO is undecidable. This result led researches to focus their study on *fragments* of FO (essentially computable sets $\mathcal{L} \subseteq \mathrm{FO}$) with the hope that expressive logics with decidable satisfiability problem could be identified.
- An important example of an interesting decidable fragment of FO is the two-variable logic FO^2 (every sentence can contain at most two variables). This logic is decidable because it has the following *bounded model property*: if $\varphi \in \mathrm{FO}^2$ is satisfiable, then it has a model of size at most $2^{|\varphi|}$.

Ordered fragments

- Recently there has been an increasing interest in studying fragments that we will refer to as the *ordered fragments* of FO.

Introduction
Ordered logic
Fluted logic
Syntactical variants
Conclusions

Ordered fragments

- Recently there has been an increasing interest in studying fragments that we will refer to as the *ordered fragments* of FO. These were originally introduced independently by Quine and Herzig.

Ordered fragments

- Recently there has been an increasing interest in studying fragments that we will refer to as the *ordered fragments* of FO. These were originally introduced independently by Quine and Herzig.
- **Main idea:** Restrict the order in which variables can be quantified, the way variables can be permuted in atomic formulas and the manner in which boolean combinations of formulas can be formed.

Ordered fragments

- Recently there has been an increasing interest in studying fragments that we will refer to as the *ordered fragments* of FO. These were originally introduced independently by Quine and Herzig.
- **Main idea:** Restrict the order in which variables can be quantified, the way variables can be permuted in atomic formulas and the manner in which boolean combinations of formulas can be formed.
- **This talk:** We will go through the syntax of the two most well-known ordered fragments (ordered logic, fluted logic) and their complexities. In addition, we will take a brief look at some recent results on the complexities of their variants (with respect to the satisfiability problem).

Syntax of ordered logic

Definition

Let $\overline{v}_\omega = (v_1, v_2, \ldots)$ be an infinite sequence of variables and let τ be a vocabulary. For every $k \in \mathbb{N}$ we define sets $\mathrm{OL}^k[\tau]$ as follows.

1. Let $R \in \tau$ be an k-ary relational symbol and consider the prefix

   $$(v_1, \ldots, v_k)$$

   of $\overline{v}_\omega$. Now $R(v_1, \ldots, v_k) \in \mathrm{OL}^k[\tau]$.
2. If $\varphi, \psi \in \mathrm{OL}^k[\tau]$, then $\neg\varphi, (\varphi \wedge \psi) \in \mathrm{OL}^k[\tau]$.
3. If $\varphi \in \mathrm{OL}^{k+1}[\tau]$, then $\exists v_{k+1}\varphi \in \mathrm{OL}^k[\tau]$.

Finally we define $\mathrm{OL}[\tau] := \bigcup_k \mathrm{OL}^k[\tau]$.

Syntax of ordered logic

Introduction
Ordered logic
Fluted logic
Syntactical variants
Conclusions

Definition

Let $\overline{v}_\omega = (v_1, v_2, \ldots)$ be an infinite sequence of variables and let τ be a vocabulary. For every $k \in \mathbb{N}$ we define sets $\mathrm{OL}^k[\tau]$ as follows.

1. Let $R \in \tau$ be an k-ary relational symbol and consider the prefix
$$(v_1, \ldots, v_k)$$
of $\overline{v}_\omega$. Now $R(v_1, \ldots, v_k) \in \mathrm{OL}^k[\tau]$.
2. If $\varphi, \psi \in \mathrm{OL}^k[\tau]$, then $\neg\varphi, (\varphi \wedge \psi) \in \mathrm{OL}^k[\tau]$.
3. If $\varphi \in \mathrm{OL}^{k+1}[\tau]$, then $\exists v_{k+1}\varphi \in \mathrm{OL}^k[\tau]$.

Finally we define $\mathrm{OL}[\tau] := \bigcup_k \mathrm{OL}^k[\tau]$.

Example

$\forall v_1(\neg P(v_1) \wedge \exists v_2 R(v_1, v_2))$ is a sentence of $\mathrm{OL}[\{P, R\}]$, while $\exists v_1 \exists v_2 R(v_2, v_1)$, $\exists v_2 \exists v_1 R(v_1, v_2)$ and $\exists v_1 \exists v_2 (P(v_2) \wedge R(v_1, v_2))$ are not.

Complexity of ordered logic

- One can prove that OL has the following bounded model property: if $\varphi \in \mathrm{OL}$ has a model, then it has one of size at most $|\varphi|$.

Complexity of ordered logic

Introduction
Ordered logic
Fluted logic
Syntactical variants
Conclusions

- One can prove that OL has the following bounded model property: if $\varphi \in \mathrm{OL}$ has a model, then it has one of size at most $|\varphi|$.
- **Main idea**: OL can't enforce that there exists more than $|\varphi|$-many elements with distinct (quantifier-free) unary types.

Complexity of ordered logic

- One can prove that OL has the following bounded model property: if $\varphi \in \mathrm{OL}$ has a model, then it has one of size at most $|\varphi|$.
- **Main idea**: OL can't enforce that there exists more than $|\varphi|$-many elements with distinct (quantifier-free) unary types.

Theorem (J.)

Over bounded vocabularies the satisfiability problem of OL *is* NP*-complete.*

Complexity of ordered logic

- One can prove that OL has the following bounded model property: if $\varphi \in \mathrm{OL}$ has a model, then it has one of size at most $|\varphi|$.
- **Main idea**: OL can't enforce that there exists more than $|\varphi|$-many elements with distinct (quantifier-free) unary types.

Theorem (J.)

Over bounded vocabularies the satisfiability problem of OL *is* NP*-complete.*

Theorem (Herzig, J.)

The satisfiability problem of OL *is* PSPACE*-complete.*

Syntax of fluted logic

Definition

Let $\overline{v}_\omega = (v_1, v_2, \dots)$ be an infinite sequence of variables and let τ be a vocabulary. For every $k \in \mathbb{N}$, we define sets $\mathrm{FL}^k[\tau]$ as follows.

1. Let $R \in \tau$ be an n-ary relation symbol and consider the subsequence
$$(v_{k-n+1}, \dots, v_k)$$
of $\overline{v}_\omega$. Now $R(v_{k-n+1}, \dots, v_k) \in \mathrm{FL}^k[\tau]$.
2. For every $\varphi, \psi \in \mathrm{FL}^k[\tau]$, we have that $\neg\varphi, (\varphi \wedge \psi) \in \mathrm{FL}^k[\tau]$.
3. If $\varphi \in \mathrm{FL}^{k+1}[\tau]$, then $\exists v_{k+1}\varphi \in \mathrm{FL}^k[\tau]$.

Finally, we define $\mathrm{FL}[\tau] := \bigcup_k \mathrm{FL}^k[\tau]$.

Syntax of fluted logic

Definition

Let $\overline{v}_\omega = (v_1, v_2, \ldots)$ be an infinite sequence of variables and let τ be a vocabulary. For every $k \in \mathbb{N}$, we define sets $\mathrm{FL}^k[\tau]$ as follows.

1. Let $R \in \tau$ be an n-ary relation symbol and consider the subsequence
$$(v_{k-n+1}, \ldots, v_k)$$
of $\overline{v}_\omega$. Now $R(v_{k-n+1}, \ldots, v_k) \in \mathrm{FL}^k[\tau]$.
2. For every $\varphi, \psi \in \mathrm{FL}^k[\tau]$, we have that $\neg\varphi, (\varphi \wedge \psi) \in \mathrm{FL}^k[\tau]$.
3. If $\varphi \in \mathrm{FL}^{k+1}[\tau]$, then $\exists v_{k+1}\varphi \in \mathrm{FL}^k[\tau]$.

Finally, we define $\mathrm{FL}[\tau] := \bigcup_k \mathrm{FL}^k[\tau]$.

Note that $\mathrm{OL} \subseteq \mathrm{FL}$.

Syntax of fluted logic

Definition

Let $\overline{v}_\omega = (v_1, v_2, \ldots)$ be an infinite sequence of variables and let τ be a vocabulary. For every $k \in \mathbb{N}$, we define sets $\mathrm{FL}^k[\tau]$ as follows.

1. Let $R \in \tau$ be an n-ary relation symbol and consider the subsequence
$$(v_{k-n+1}, \ldots, v_k)$$
of $\overline{v}_\omega$. Now $R(v_{k-n+1}, \ldots, v_k) \in \mathrm{FL}^k[\tau]$.
2. For every $\varphi, \psi \in \mathrm{FL}^k[\tau]$, we have that $\neg\varphi, (\varphi \wedge \psi) \in \mathrm{FL}^k[\tau]$.
3. If $\varphi \in \mathrm{FL}^{k+1}[\tau]$, then $\exists v_{k+1}\varphi \in \mathrm{FL}^k[\tau]$.

Finally, we define $\mathrm{FL}[\tau] := \bigcup_k \mathrm{FL}^k[\tau]$.

Note that $\mathrm{OL} \subseteq \mathrm{FL}$.

Example

$\exists v_1 \exists v_2 (P(v_2) \wedge R(v_1, v_2))$ is a sentence of $\mathrm{FL}[\{P, R\}]$, while $\exists v_1 \exists v_2 \exists v_3 (R(v_1, v_2) \wedge R(v_2, v_3))$ is not.

Complexity of fluted logic

- FL also has a bounded model property: if $\varphi \in \mathrm{FL}$ has a model, then it has one of size at most

$$\underbrace{2^{2^{\cdot^{\cdot^{2}}}}}_{|\varphi|^k\text{-times}},$$

for some constant k.

Complexity of fluted logic

- FL also has a bounded model property: if $\varphi \in \mathrm{FL}$ has a model, then it has one of size at most

$$\underbrace{2^{2^{\cdot^{\cdot^{2}}}}}_{|\varphi|^k\text{-times}},$$

  for some constant k.

- **Main idea of proof**: For each $k \geq 2$ and $\varphi \in \mathrm{FL}^{(k+1)}$, there exists $\varphi' \in \mathrm{FL}^k$ so that φ has a model iff φ' has, $|\varphi'| = 2^{O(|\varphi|)}$ and if φ' has a model of size N, then φ has a model of size at most $|\varphi|N$.

Complexity of fluted logic

Introduction
Ordered logic
Fluted logic
Syntactical variants
Conclusions

- FL also has a bounded model property: if $\varphi \in \mathrm{FL}$ has a model, then it has one of size at most

$$\underbrace{2^{2^{\cdot^{\cdot^{2}}}}}_{|\varphi|^k\text{-times}},$$

for some constant k.

- **Main idea of proof**: For each $k \geq 2$ and $\varphi \in \mathrm{FL}^{(k+1)}$, there exists $\varphi' \in \mathrm{FL}^k$ so that φ has a model iff φ' has, $|\varphi'| = 2^{O(|\varphi|)}$ and if φ' has a model of size N, then φ has a model of size at most $|\varphi|N$.

Theorem (Pratt-Hartmann, Swast, Tendera)

The satisfiability problem of FL *is* $\textsc{Tower}$*-complete.*

Algebraic view on ordered fragments

- Originally some of the ordered fragments were discovered by Quine as a by-product of his (eventually successful) attempt at giving a variable-free syntax for FO (predicate functor logic(s)).

Algebraic view on ordered fragments

- Originally some of the ordered fragments were discovered by Quine as a by-product of his (eventually successful) attempt at giving a variable-free syntax for FO (predicate functor logic(s)).
- OL can be seen as consisting of three (relational) algebraic operators: complement $\neg$, intersection $\cap$ and projection $\exists$.

Algebraic view on ordered fragments

- Originally some of the ordered fragments were discovered by Quine as a by-product of his (eventually successful) attempt at giving a variable-free syntax for FO (predicate functor logic(s)).
- OL can be seen as consisting of three (relational) algebraic operators: complement $\neg$, intersection $\cap$ and projection $\exists$.
- Similarly, FL consists of $\neg, \exists$ and the so-called suffix intersection $\dot{\cap}$ (allows one to compute intersections of relations that have different arities).

Algebraic view on ordered fragments

- Originally some of the ordered fragments were discovered by Quine as a by-product of his (eventually successful) attempt at giving a variable-free syntax for FO (predicate functor logic(s)).
- OL can be seen as consisting of three (relational) algebraic operators: complement $\neg$, intersection $\cap$ and projection $\exists$.
- Similarly, FL consists of $\neg, \exists$ and the so-called suffix intersection $\dot{\cap}$ (allows one to compute intersections of relations that have different arities).
- This point of view suggests naturally several syntactical variants of, say, OL and FL (simply add or remove algebraic operators).

Brief summary of recent complexity results

- Adding (restricted) use of equality either to OL or FL does not affect complexity.

Brief summary of recent complexity results

- Adding (restricted) use of equality either to OL or FL does not affect complexity.
- Adding a swap operator (swap the last two elements in every tuple) increases complexity significantly: OL becomes NExpTime-complete while FL becomes undecidable.

Brief summary of recent complexity results

- Adding (restricted) use of equality either to OL or FL does not affect complexity.
- Adding a swap operator (swap the last two elements in every tuple) increases complexity significantly: OL becomes $\mathrm{NE{\scriptsize XP}T{\scriptsize IME}}$-complete while FL becomes undecidable.
- Replacing $\exists$ with one-dimensional quantification (select at most the first element from every tuple) decreases complexity: OL becomes NP-complete while FL becomes $\mathrm{NE{\scriptsize XP}T{\scriptsize IME}}$-complete. One-dimensional FL remains $\mathrm{NE{\scriptsize XP}T{\scriptsize IME}}$-complete even in the presence of equality and swap.

Conclusions

- Ordered fragments present a fresh viewpoint on the question of what makes satisfiability problems decidable (feasible).

Conclusions

- Ordered fragments present a fresh viewpoint on the question of what makes satisfiability problems decidable (feasible).
- Looking at intersections of ordered fragments with *guarded fragments* seems to be a very promising research direction (modal logics often have variable-free syntax).

Conclusions

- Ordered fragments present a fresh viewpoint on the question of what makes satisfiability problems decidable (feasible).
- Looking at intersections of ordered fragments with *guarded fragments* seems to be a very promising research direction (modal logics often have variable-free syntax).
- How much can we extend the expressive power of the fluted logic while preserving its decidability?

Conclusions

Introduction
Ordered logic
Fluted logic
Syntactical variants
Conclusions

- Ordered fragments present a fresh viewpoint on the question of what makes satisfiability problems decidable (feasible).
- Looking at intersections of ordered fragments with *guarded fragments* seems to be a very promising research direction (modal logics often have variable-free syntax).
- How much can we extend the expressive power of the fluted logic while preserving its decidability?

Thanks! :-)