A step up in expressiveness of decidable fixpoint logics

Michael Benedikt$^1$, Pierre Bourhis$^2$, and Michael Vanden Boom$^1$

$^1$University of Oxford
$^2$CNRS CRIStAL, Université Lille 1, INRIA Lille

LICS 2016
New York, USA
Fixpoint logics can express dynamic, recursive properties.

Example

binary relation $R$, unary relation $P$

“from $w$, it is possible to $R$-reach some $P$-element”

$[\text{Reach}-P](w)$
Fixpoint logics can express dynamic, recursive properties.

Example

binary relation $R$, unary relation $P$

“from $w$, it is possible to $R$-reach some $P$-element”

$\[ \text{lfp}_{y,y} . \exists z (Ryz \land (Pz \lor Yz)) \](w) \]
**LFP**

**LFP**: extension of first-order logic with fixpoint formulas \([\text{lfp}_{y,Y}.\psi(y,Y)](w)\) for \(\psi(y,Y)\) positive in \(Y\) (of arity \(m = |y|\)).

For all structures \(\mathcal{A}\), the formula \(\psi\) induces a monotone operation

\[
P(A^m) \longrightarrow P(A^m)
\]

\[
V \longmapsto \psi_{\mathcal{A}}(V) := \{a \in A^m : \mathcal{A}, a, V \models \psi\}
\]

⇒ there is a unique **least fixpoint** \([\text{lfp}_{y,Y}.\psi(y,Y)]_{\mathcal{A}} := \bigcup_a \psi_{\mathcal{A}}^a\)

\[
\psi_{\mathcal{A}}^0 := \emptyset
\]

\[
\psi_{\mathcal{A}}^{a+1} := \psi_{\mathcal{A}}(\psi_{\mathcal{A}}^a)
\]

\[
\psi_{\mathcal{A}}^\lambda := \bigcup_{a<\lambda} \psi_{\mathcal{A}}^a
\]
LFP: extension of first-order logic with fixpoint formulas \([\text{lfp}_{y,y} \cdot \psi(y, Y)](w)\) for \(\psi(y, Y)\) positive in \(Y\) (of arity \(m = |y|\)).

For all structures \(\mathfrak{A}\), the formula \(\psi\) induces a monotone operation

\[
P(A^m) \rightarrow P(A^m)
\]

\[
V \mapsto \psi_{\mathfrak{A}}(V) := \{a \in A^m : \mathfrak{A}, a, V \models \psi\}
\]

⇒ there is a unique least fixpoint \([\text{lfp}_{y,y} \cdot \psi(y, Y)]_{\mathfrak{A}} := \bigcup_a \psi^a_{\mathfrak{A}}\)

\[
\begin{align*}
\psi^0_{\mathfrak{A}} & := \emptyset \\
\psi^{a+1}_{\mathfrak{A}} & := \psi_{\mathfrak{A}}(\psi^a_{\mathfrak{A}}) \\
\psi^\lambda_{\mathfrak{A}} & := \bigcup_{a < \lambda} \psi^a_{\mathfrak{A}}
\end{align*}
\]

Semantics of fixpoint operator: \(\mathfrak{A}, a \models [\text{lfp}_{y,y} \cdot \psi(y, Y)](w)\) iff \(a \in \bigcup_a \psi^a_{\mathfrak{A}}\)
Examples

“from $w$, it is possible to $R$-reach some $P$-element”

$$\left[ \text{lfp}_{Y,y} . \exists z ( Ryz \land (Pz \lor Yz)) \right](w)$$

$\xrightarrow{}$ $\xrightarrow{}$ $\xrightarrow{}$ $\xrightarrow{}$ $\xrightarrow{}$ $\xrightarrow{}$ $\xrightarrow{}$ $\xrightarrow{}$ $\xrightarrow{}$ $\xrightarrow{}$
Examples

“from \( w \), it is possible to \( R \)-reach some \( P \)-element”

\[
[lfp_{y,y} \cdot \exists z (Ryz \land (Pz \lor Yz))] (w)
\]

\[
\xymatrix{a_1 \ar[r] & a_2 \ar[r] & a_3 \ar[r] & \cdots \ar[r] & a_k \ar[r] & a_{k+1}}
\]

“from \( w \), it is possible to \( R \)-reach \( x \)”, i.e. “\((w, x)\) is in the transitive closure of \( R \)”

\[
[lfp_{y,y} \cdot \exists z (Ryz \land (z = x \lor Yz))] (w)
\]

(Free first-order variable \( x \) in the fixpoint predicate is called a parameter.)
The family of “guarded” fixpoint logics has decidable satisfiability.

Guarded fixpoint logic (GFP): Andréka, van Benthem, Németi ’95-’98; Grädel, Walukiewicz ’99
Unary negation fixpoint logic (UNFP): ten Cate, Segoufin ’11
Guarded negation fixpoint logic (GNFP): Bárány, ten Cate, Segoufin ’11
Guarded negation fixpoint logic (GNFP)

Let \( \sigma \) be a signature of relations and constants.

**Syntax of GNFP[\( \sigma \)]**

\[
\phi ::= R t \mid Y t \mid \phi \land \phi \mid \phi \lor \phi \mid \exists y(\psi(x)) \mid G(x) \land \neg\psi(x) \mid \\
\left[ \text{lfp}_{Y,y} . G(y) \land \phi(y, Y, Z) \right](t) \quad \text{where } Y \text{ only occurs positively in } \phi
\]

where \( R \) and \( G \) are relations in \( \sigma \) or =, and \( t \) is a tuple over variables and constants.
Guarded negation fixpoint logic (GNFP)

Let $\sigma$ be a signature of relations and constants.

**Syntax of GNFP[\sigma]**

$$
\varphi ::= R \, t \mid Y \, t \mid \varphi \land \varphi \mid \varphi \lor \varphi \mid \exists y (\psi(x)) \mid G(x) \land \neg \psi(x) \mid \\
[\text{lfp}_{Y,y} . G(y) \land \varphi(y, Y, Z)](t) \quad \text{where } Y \text{ only occurs positively in } \varphi
$$

where $R$ and $G$ are relations in $\sigma$ or $=$, and $t$ is a tuple over variables and constants.

**Restrictions on fixpoint operator:**

- must define a guarded relation  
  (tuples in the fixpoint must be guarded by an atom from $\sigma$ or $=$)  
- cannot use parameters
These guarded fixpoint logics all have the **tree-like model property** (models with tree decompositions of bounded tree-width)

⇒ amenable to **tree automata techniques**
Satisfiability

These guarded fixpoint logics all have the tree-like model property (models with tree decompositions of bounded tree-width)

⇒ amenable to tree automata techniques

**Theorem** (Grädel, Walukiewicz ’99; Bárány, Segoufin, ten Cate ’11; Bárány, Bojańczyk ’12)

Satisfiability and finite satisfiability are decidable for guarded fixpoint logics (2EXPTIME in general, EXPTIME for fixed-width formulas in GFP).

**Idea:** Reduce to tree automaton emptiness test.
Examples

In GNFP:

\[ \text{lfp}_{Y,y} \cdot \exists z (Ryz \land (Pz \lor Yz)) \](w)
Examples

In GNFP:

$$\left[ \text{Ifp}_{y,y} \cdot y = y \land \exists z (Ryz \land (Pz \lor Yz)) \right](w)$$
Examples

In GNFP:

$$[\text{lfp}_{Y,y} \cdot y = y \land \exists z (Ryz \land (Pz \lor Yz))] (w)$$

Not in GNFP:

$$[\text{lfp}_{Y,y} \cdot y = y \land \exists z (Ryz \land (z = x \lor Yz))] (w)$$
Can we go further?

Recall the restrictions on the fixpoint operators in GNFP:

- must define a guarded relation
- cannot use parameters

Which of these restrictions are essential for decidability?
Can we go further?

Recall the restrictions on the fixpoint operators in GNFP:

- must define a guarded relation
- cannot use parameters

Which of these restrictions are essential for decidability?

**Answer:** only first one!
GNFP^UP: extend GNFP with *unguarded parameters in fixpoint*
**GNFP\textsuperscript{UP}**

**GNFP\textsuperscript{UP}: extend GNFP with unguarded parameters in fixpoint**

### Syntax of GNFP\textsuperscript{UP}[\sigma]

\[
\varphi ::= R t \mid Y t \mid \varphi \land \varphi \mid \varphi \lor \varphi \mid \exists y (\psi (x y)) \mid G (x) \land \neg \psi (x) \mid \\
[ \text{Ifp}_{Y, y} . G (y) \land \varphi (x, y, Y, Z)](t) \text{ where } Y \text{ only occurs positively in } \varphi
\]

where \( R \) and \( G \) are relations in \( \sigma \) or \( = \), and \( t \) is a tuple over variables and constants.
GNFP<sup>UP</sup> : extend GNFP with unguarded parameters in fixpoint

**Syntax of GNFP<sup>UP</sup>[σ]**

\[
\varphi ::= R t \mid Y t \mid \varphi \land \varphi \mid \varphi \lor \varphi \mid \exists y(\psi(x y)) \mid G(x) \land \neg \psi(x) \mid \\
\left[\text{lfp}_{Y,y} \cdot G(y) \land \varphi(x, y, Y, Z)(t)\right] (t) \quad \text{where } Y \text{ only occurs positively in } \varphi
\]

where \( R \) and \( G \) are relations in \( \sigma \) or \( = \), and \( t \) is a tuple over variables and constants.

**Example**

GNFP<sup>UP</sup> can express the transitive closure of a binary relation \( R \) using

\[
\left[\text{lfp}_{Y,y} \cdot \exists z (Ryz \land (z = x \lor Yz))\right](w)
\]
Expressivity of GNFP$^\text{UP}$

$\exists xyz \left( \left[ R^* S \right](x, y) \land \left[ S \mid R \right](y, z) \land P(z) \right)$

GNFP$^\text{UP}$ also subsumes

C2RPQs (conjunctive 2-way regular path queries)

MQs and GQs [Rudolph, Krötzsch ‘13 ; Bourhis, Krötzsch, Rudolph ‘15]
Satisfiability for $\text{GNFP}^{\text{Up}}$

$\text{GNFP}^{\text{Up}}$ still has tree-like models
$\Rightarrow$ still amenable to tree automata techniques

Unlike other guarded logics, satisfiability testing for $\varphi \in \text{GNFP}^{\text{Up}}$ is non-elementary, with running time

$$2^{2^{f(|\varphi|)}}$$

where the height of the tower depends only on the parameter depth: the number of nested parameter changes in the formula.
Satisfiability for $\text{GNFP}^{\text{UP}}$

$\text{GNFP}^{\text{UP}}$ still has tree-like models
⇒ still amenable to tree automata techniques

Unlike other guarded logics, satisfiability testing for $\varphi \in \text{GNFP}^{\text{UP}}$ is non-elementary, with running time

$$2^{2^{\cdots 2^{f(|\varphi|)}}}$$

where the height of the tower depends only on the parameter depth: the number of nested parameter changes in the formula.

**Theorem**

Satisfiability is decidable for $\varphi \in \text{GNFP}^{\text{UP}}$ in $(n + 2)$-EXPTIME, where $n$ is the parameter depth of $\varphi$. 
It is known that satisfiability is undecidable for GFP (even without fixpoints) when certain relations are required to be transitive.

[Grädel ‘99, Ganzinger et al. ‘99]
Skirting undecidability

It is known that satisfiability is undecidable for GFP (even without fixpoints) when certain relations are required to be transitive. [Grädel ‘99, Ganzinger et al. ‘99]

$\text{GNFP}^{\text{UP}}$ can express the transitive closure of a binary relation $R$ using

$$[\text{Ifp}_{y,y}. \exists z (Ryz \land (z = x \lor Yz))] (w).$$

But it cannot enforce that $R$ is transitive.
# FO-definability

<table>
<thead>
<tr>
<th>Theorem</th>
</tr>
</thead>
<tbody>
<tr>
<td>It is decidable whether $<a href="w">\text{Ifp}_{y,y} . G(y) \land \psi(x, y, Y)</a> \in \text{GNFP}^{\text{UP}}$ can be expressed in FO (when $\psi$ does not use any additional fixpoints).</td>
</tr>
</tbody>
</table>

It is decidable whether a C2RPQ can be expressed in FO.

**Idea:** Adapt automata for $\text{GNFP}^{\text{UP}}$, and reduce to a boundedness question for cost automata (automata with counters).
Conclusion

We can allow unguarded parameters in guarded fixpoint logics.

Contributions

We showed that:

- tree automata techniques can be used to analyze $\text{GNFP}^\text{UP}$
- satisfiability is decidable for $\text{GNFP}^\text{UP}$, and the key factor impacting the complexity is the parameter depth
- some boundedness and FO-definability problems are decidable for $\text{GNFP}^\text{UP}$
Conclusion

We can allow unguarded parameters in guarded fixpoint logics.

Contributions

We showed that:

- tree automata techniques can be used to analyze GNFP\textsuperscript{UP}
- satisfiability is decidable for GNFP\textsuperscript{UP}, and the key factor impacting the complexity is the parameter depth
- some boundedness and FO-definability problems are decidable for GNFP\textsuperscript{UP}

Open question

Is finite satisfiability decidable for GNFP\textsuperscript{UP}?