Computing Nested Fixpoints in Quasipolynomial Time

Daniel Hausmann and Lutz Schröder
LaBRI – October 22 2019

Chair for Theoretical Computer Science
Friedrich-Alexander Universität Erlangen-Nürnberg, Germany
Why Nested Fixpoints?

- **Model checking** for the $\mu$-calculus = solving parity games.
- **Satisfiability checking** for the $\mu$-calculus by solving parity games.
- Winning regions of parity games are **nested fixpoints**.
- Model checking and satisfiability checking for generalized $\mu$-calculi (graded, probabilistic, alternating-time) by nested fixpoints.
- **Synthesis** for linear-time logics (e.g. LTL).
- Computing generalized **fair bisimulations**.
- **Type checking** for inductive-coinductive types.

We show:

- Nested fixpoints stabilize after quasipolynomially many iterations.
- The problem of computing nested fixpoints is in $\text{NP} \cap \text{co-NP}$.
- Zielonka's algorithm can be adapted to compute nested fixpoints.
Why Nested Fixpoints?

- Model checking for the $\mu$-calculus = solving parity games.
- Satisfiability checking for the $\mu$-calculus by solving parity games.
- Winning regions of parity games are nested fixpoints.
- Model checking and satisfiability checking for generalized $\mu$-calculi (graded, probabilistic, alternating-time) by nested fixpoints.
- Synthesis for linear-time logics (e.g. LTL).
- Computing generalized fair bisimulations.
- Type checking for inductive-coinductive types.

We show:

- Nested fixpoints stabilize after quasipolynomially many iterations.
- The problem of computing nested fixpoints is in $\text{NP} \cap \text{co-NP}$.
- Zielonka’s algorithm can be adapted to compute nested fixpoints.
Function $\alpha : \mathcal{P}(U)^k \to \mathcal{P}(U)$ is monotone if for all $U_i \subseteq V_i$, $1 \leq i \leq k$,

$$\alpha(U_1, \ldots, U_k) \subseteq \alpha(V_1, \ldots, V_k)$$

**Extremal Fixpoints, Nested Fixpoints**

Let $f : \mathcal{P}(U) \to \mathcal{P}(U)$ and $\alpha : \mathcal{P}(U)^k \to \mathcal{P}(U)$ be monotone functions.

- $\text{LFP } f = \bigcap \{Z \subseteq U \mid f(Z) \subseteq Z\}$
- $\text{GFP } f = \bigcup \{Z \subseteq U \mid Z \subseteq f(Z)\}$
- $\text{NFP } \alpha = \eta_k X_k \cdot \eta_{k-1} X_{k-1} \ldots \cdot \eta_1 X_1 \cdot \alpha(X_1, \ldots, X_k)$,

where $\eta_i = \text{LFP}$ if $i$ is odd, $\eta_i = \text{GFP}$ if $i$ is even.
Parity game \((V = V_\exists \cup V_\forall, E \subseteq V \times V, \Omega)\) with \(k\) priorities. Define:

\[
\Omega_i = \{v \in V \mid \Omega(v) = i\}
\]

\[
\Diamond U = \{v \in V \mid E(v) \cap U \neq \emptyset\}
\]

\[
\Box U = \{v \in V \mid E(v) \subseteq U\}
\]

\[
\alpha_{PG}(X_1, \ldots, X_k) = (V_\exists \cap (\bigcup_{1 \leq i \leq k} \Omega_i \cap \Diamond X_i)) \cup (V_\forall \cap (\bigcup_{1 \leq i \leq k} \Omega_i \cap \Box X_i))
\]

**Theorem (e.g. [Dawar,Grädel,2008],[Bruse,Falk,Lange,2014])**

\[
\text{win}_\exists = \text{NFP} \alpha_{PG}
\]
A Tool: Fixpoint Parity Games (Venema, König et al.)

Fixpoint Parity Game for \(\text{NFP}_\alpha\)

Parity game \((V, E, \Omega)\), nodes: \(V = U \cup \mathcal{P}(U)^k \cup \mathcal{P}(U) \times \{1, \ldots, k\}\)

| node     | priority | owner | moves to                                      |
|----------|----------|-------|-----------------------------------------------|
| \(u \in U\) | 0        | \(\exists\) | \(\{U \in \mathcal{P}(U)^k | u \in \alpha(U)\}\) |
| \(U\)    | 0        | \(\forall\) | \(\{(U_j, j) | 1 \leq j \leq k\}\)        |
| \((U, j)\) | \(j\)   | \(\forall\) | \(\{v | v \in U\}\)                  |

where \(U = (U_1, \ldots, U_k) \in \mathcal{P}(U)^k\).

Theorem [König et al. 2019]

Eloise wins node \(u\) if and only if \(u \in \text{NFP}_\alpha\).

Problem: exponential size
- still useful for showing *history-freeness* for nested fixpoints.
History-freeness for Nested Fixpoints

History-free witnesses

Even graph $S \subseteq U \times \{1, \ldots, k\} \times U$ s.t. for all $(u, p, u') \in S$,

$$u \in \alpha(S_1(u), \ldots, S_k(u)),$$

where $S_i(u) = \{v \mid (u, i, v) \in S\}$.

Note: $|S| \in O(|U|^2)$

Lemma

There is a history-free witness mentioning $u$ if and only if $u \in \text{NFP} \alpha$. 
Theorem

If \( \alpha(X_1, \ldots, X_k) \) can, for all \( X_1, \ldots, X_n \), be computed in polynomial time, the problem of computing NFP \( \alpha \) is in \( \text{NP} \cap \text{co-NP} \).

Proof: Each State is contained in NFP or in dual nested fixpoint, hence containment in \( \text{NP} \) suffices. Guess *polynomial*-sized history-free witness for Eloise winning exponential-sized game. Verify witness in polynomial time: check that all paths are even and verify compatibility with \( \alpha \).
Idea: Annotate nodes with quasipolynomial histories ("statistics")

$$\bar{\sigma} = (o_{\lceil \log n \rceil + 1}, \ldots, o_0) \quad 1 \leq o_i \leq k$$

Define $\bar{\sigma}@i = (o'_{\lceil \log n \rceil + 1}, \ldots, o'_0)$ as follows:

- $i$ even: pick greatest $j$ s.t. $i > o_j > 0$. If no such $j$ exists, then $j = \ast$.
- $i$ odd: pick greatest $j$ s.t.
  - a) $i > o_j > 0$ or
  - b) $o_j$ even for all $j' < j$, $o_{j'}$ odd (and if $o_j > 0$, $i < o_j$).
- If $j = \ast$, then $\bar{\sigma}@i = \bar{\sigma}$. Otherwise, $o'_{j'} = o_{j'}$ for $j' > j$, $o'_i = i$ and $o'_{j'} = 0$ for $j' < j$.

Move from $(v, \bar{\sigma})$ to $(w, \bar{\sigma}@\Omega(w))$ if move from $v$ to $w$ exists in original game. Solve safety game of quasipolynomial size $n \cdot k^{\lceil \log n \rceil + 2}$. 
Use Calude et al.’s quasipolynomial histories to compute nested fixpoint:

Put $hi = \{(o_{\lceil \log n \rceil + 1}, \ldots, o_0) \mid 1 \leq o_i \leq k\}$ having $|hi| \leq k^{\lceil \log n \rceil + 2}$ and define $\gamma : \mathcal{P}(U \times hi) \to \mathcal{P}(U \times hi)$ by

$$\gamma(Y) = \{(v, \bar{o}) \in (U \times hi) \mid v \in \alpha(\bar{Y}^{\bar{o}^{@1}}, \ldots, \bar{Y}^{\bar{o}^{@k}})\}$$

where

$$\bar{Y}^{\bar{o}'} = \begin{cases} \emptyset & \text{leftmost digit in } \bar{o}' \text{ is not } 0 \\ \{u \in U \mid (u, \bar{o}') \in Y\} & \text{otherwise.} \end{cases}$$

**Main Theorem:**

Let $\alpha : \mathcal{P}(U)^k \to \mathcal{P}(U)$ be monotone. Then $\text{NFP } \alpha = \pi_1[\text{GFP } \gamma]$. 
Zielonka’s Algorithm for Solving Parity Games

Define

\[ \text{Attr}_{\exists}^\text{PG}(G, F) = \mu X. G \cap (F \cup \alpha_{\text{PG}}(X, \ldots, X)) \]
\[ \text{Attr}_{\forall}^\text{PG}(G, F) = \mu X. G \cap (F \cup \overline{\alpha_{\text{PG}}}(X, \ldots, X)) \]

Algorithm: Solve parity game \((G, E, \Omega)\) [Zielonka]

1. **procedure** `SOLVE_\exists(G, i)` \(\triangleright i\) even
2. \(N_i := \{v \in G \mid \Omega(v) = i\}\); \(\triangleright\) maximal priority nodes
3. \(H := G \setminus \text{Attr}_{\exists}^\text{PG}(G, N_i)\); \(\triangleright\) exclude Eloise-attractor of \(N_i\)
4. \(W_\forall := \text{SOLVE}_\forall(H, i - 1)\); \(\triangleright\) solve smaller game
5. \(G := G \setminus \text{Attr}_{\forall}^\text{PG}(G, W_\forall)\); \(\triangleright\) remove Abelard-attractor of \(W_\forall\)
6. **if** \(W_\forall \neq \emptyset\) **then** GOTO 2:
7. **else** RETURN \(G\).
Zielonka’s Algorithm for Computing Nested Fixpoints

Define

\[ \text{Attr}_\exists (G, F) = \mu X. G \cap (F \cup \alpha(X, \ldots, X)) \]
\[ \text{Attr}_\forall (G, F) = \mu X. G \cap (F \cup \overline{\alpha}(X, \ldots, X)) \]

Algorithm: Compute NFP \( \alpha \)

1: procedure \( \text{SOLVE}_\exists (G, i) \) \( \triangleright i \) even
2: \( N_i := \{ v \in G \mid \Omega(v) = i \} \); \( \triangleright \) maximal priority nodes
3: \( H := G \setminus \text{Attr}_\exists (G, N_i) \); \( \triangleright \) exclude Eloise-attractor of \( N_i \)
4: \( W_\forall := \text{SOLVE}_\forall (H, i - 1) \); \( \triangleright \) compute smaller fixpoint
5: \( G := G \setminus \text{Attr}_\forall (G, W_\forall) \); \( \triangleright \) remove Abelard-attractor of \( W_\forall \)
6: if \( W_\forall \neq \emptyset \) then GOTO 2:
7: else RETURN \( G \).
The Fixpoint Law behind Zielonka’s Algorithm

NFP $\alpha$ as a system of equations:

\[
\begin{align*}
X_i &= \text{LFP } X_{i-1} & i > 1, i \text{ odd} \\
X_i &= \text{GFP } X_{i-1} & i \text{ even} \\
X_1 &= \text{GFP } \alpha(X_1, \ldots, X_k)
\end{align*}
\]

A second system of equations:

\[
\begin{align*}
Y_i &= \text{LFP } (\Omega > (i) \cup \alpha(Y_i, \ldots, Y_i) \cup Y_{i-1}) \cap (\Omega \leq (i) \cup Y_{i+1}) & i \text{ odd} \\
Y_i &= \text{GFP } (\Omega \leq (i) \cap \alpha(Y_i, \ldots, Y_i) \cap Y_{i-1}) \cup (\Omega > (i) \cap Y_{i+1}) & i \text{ even}
\end{align*}
\]

**Theorem:**

$X_k = Y_k.$
Set $V$ of fixpoint variables, set $\Lambda$ of modalities, closed under duals.

**Syntax:**

\[
\phi, \psi := \top | \bot | \phi \land \psi | \phi \lor \psi | X | \Diamond \psi | \mu X. \psi | \nu X. \psi \quad \Diamond \in \Lambda, X \in V
\]

**Set-endofunctor $T$, predicate lifting**\(^1\) for $\Diamond \in \Lambda$: natural transformation

\[
[\Diamond] : Q \to Q \circ T^{op}
\]

E.g. for $T = \mathcal{P}$,

\[
[\Diamond](A) = \{B \in \mathcal{P}(C) | B \cap A \neq \emptyset\}
\]

\[
[\Box](A) = \{B \in \mathcal{P}(C) | B \subseteq A\}
\]

---

\(^1\)[Pattinson, 2007]
The Coalgebraic $\mu$-Calculus [Cîrstea et al., 2011]

Assume monotonicity of predicate liftings ($A \subseteq B \Rightarrow [\Diamond]A \subseteq [\Diamond]B$)

**Semantics:**

Models: $T$-coalgebras $(C, \xi : C \to TC)$, extension of formulas:

$\begin{align*}
[X]_\sigma &= \sigma(X) \\
[\mu X. \psi]_\sigma &= \text{LFP}([\psi]^{X}_\sigma) \\
[\nu X. \psi]_\sigma &= \text{GFP}([\psi]^{X}_\sigma)
\end{align*}$

where $\sigma : V \to \mathcal{P}(C)$, where $[\psi]^{X}_\sigma(A) = [\psi]_{\sigma[X\mapsto A]}$ for $A \subseteq C$ and where $(\sigma[X\mapsto A])(X) = A$, $(\sigma[X\mapsto A])(Y) = \sigma(Y)$ for $X \neq Y$. 
Instances of the Coalgebraic $\mu$-Calculus

- $T = \mathcal{P}$: transition systems $(C, \xi : C \to \mathcal{P}(C))$
  - modalities: $\Diamond, \Box$
  - standard $\mu$-calculus, e.g. $\mu X. \psi \lor \Box X$

- $T = B$ (bag functor): graded transition systems $(C, \xi : C \to \mathcal{B}(C))$
  - modalities: $\langle g \rangle, [g], g \in \mathbb{N}$
  - graded $\mu$-calculus$^2$, e.g. $\mu X. \psi \lor \langle 1 \rangle X$

- $T = \mathcal{G}$: concurrent game frames
  - Set $N$ of agents, modalities $[D], \langle D \rangle, D \subseteq N$
  - alternating-time $\mu$-calculus$^3$, e.g. $\nu X. \psi \land [D]X$

- $T = \mathcal{D}$: Markov chains
  - modalities $\langle p \rangle, [p], p \in \mathbb{Q} \cap [0, 1]$
  - (two-valued) probabilistic $\mu$-calculus, e.g. $\nu X. \psi \land \langle 0.5 \rangle X$

---

$^2$[Kupferman et al., 2002]  
$^3$[Alur et al., 2002]
Recent Results on the Coalgebraic $\mu$-Calculus

- Reduce model checking [H, Schröder, CONCUR 2019] and satisfiability checking [H, Schröder, FoSSaCS 2019] for the coalgebraic $\mu$-calculus to computing nested fixpoints.

**Corollary**

Model checking for coalgebraic $\mu$-calculi is in $\text{QP}$ and in $\text{NP} \cap \text{Co-NP}$.

**Corollary**

Satisfiability checking for coalgebraic $\mu$-calculi can be done in time $\mathcal{O}(2^{nk \log n})$ (down from $\mathcal{O}(2^{n^2 k^2 \log n})$).
Introducing: Coalgebraic Parity Games

**Definition - Coalgebraic parity game:**

A $T$-coalgebra $(C, \xi : C \to TC)$ with mappings $\Omega : C \to \mathbb{N}$, $m : C \to \Lambda$.

Eloise wins node $c \in C$ if there is an even graph $(D, R)$ on $C$ such that:

$$\text{for all } d \in D, \xi(d) \in \llbracket m(d) \rrbracket R(d).$$
**Definition - Coalgebraic parity game:**

$T$-coalgebra $(C, \xi : C \to TC)$ with mappings $\Omega : C \to \mathbb{N}$, $m : C \to \Lambda$.

Eloise wins node $c \in C$ if there is even graph $(D, R)$ on $C$ s.t.

$$\text{for all } d \in D, \xi(d) \in \llbracket m(d) \rrbracket R(d).$$

E.g.

- $T = P$: parity game for $T$ is graph $(C, \xi : C \to \mathcal{P}(C))$ with priority map $\Omega$ and node ownership map $m : C \to \\{\Diamond, \Box\}$. 
**Definition - Coalgebraic parity game:**

T-coalgebra \((C, \xi : C \to TC)\) with mappings \(\Omega : C \to \mathbb{N}, m : C \to \Lambda\).

Eloise wins node \(c \in C\) if there is even graph \((D, R)\) on \(C\) s.t.

\[
\text{for all } d \in D, \xi(d) \in [m(d)]R(d).
\]

**e.g.**

- \(T = \mathcal{P}\): parity game for \(T\) is graph \((C, \xi : C \to \mathcal{P}(C))\) with priority map \(\Omega\) and node ownership map \(m : C \to \{\Diamond, \Box\}\).

- \(T = \mathcal{D}\): parity game for \(T\) is Markov chain \((C, \xi : C \to \mathcal{D}(C))\) with priority map \(\Omega\) and map \(m : C \to \{\langle p\rangle, [p] \mid p \in \mathbb{Q} \cap [0, 1]\}\).
Coalgebraic Parity Games, examples

$T = \mathcal{P}$: standard

$T = \mathcal{B}$: graded

$T = \mathcal{D}$: probabilistic
Coalgebraic Parity Games, examples, strategies

\( T = \mathcal{P} \): standard

\( T = \mathcal{B} \): graded

\( T = \mathcal{D} \): probabilistic
Winning regions in coalgebraic parity games are nested fixpoints:

Given game $(C, \xi, m, \Omega)$, define $f : \mathcal{P}(C)^k \rightarrow \mathcal{P}(C)$ by

$$f(X_0, \ldots, X_k) = \{v \mid \exists i, \heartsuit \in \Lambda. m(v) = \heartsuit, \Omega(v) = i \text{ and } \xi(v) \in \llbracket \heartsuit \rrbracket X_i\}$$
Winning regions in coalgebraic parity games are nested fixpoints:

Given game \((C, \xi, m, \Omega)\), define \(f : \mathcal{P}(C)^k \rightarrow \mathcal{P}(C)\) by

\[
f(X_0, \ldots, X_k) = \{ v \mid \exists i, \Diamond \in \Lambda. m(v) = \Diamond, \Omega(v) = i \text{ and } \xi(v) \in \lbrack \Diamond \rbrack X_i \}\]

**Theorem [H, Schröder, CONCUR 2019]**:

Player Eloise wins \(u\) in coalgebraic parity game if and only if \(u \in \text{NFP } f\).

Coalgebraic \(\mu\)-calculus model checking = solving coalgebraic parity games.
Enables on-the-fly model checking: Start with initial node, expand nodes step by step, compute NFP \(f\) at any point (solving a partial game).
Results:

– Computing nested fixpoints by
  • (fixpoint iteration),
  • Calude et al.’s quasipolynomial algorithm
  • Zielonka’s algorithm
– Computing nested fixpoints also is in $\text{NP} \cap \text{Co-NP}$.
– Reduction of satisfiability checking and model checking for the coalgebraic $\mu$-calculus to computing nested fixpoints.

Future work:

– Computing fair bisimulations as nested fixpoints.
– Type checking for inductive-coinductive types by computing nested fixpoints.
R. Alur, T. Henzinger, and O. Kupferman.  
**Alternating-time temporal logic.**  
*J. ACM*, 49:672–713, 2002.

F. Bruse, M. Falk, and M. Lange.  
**The fixpoint-iteration algorithm for parity games.**  
In *Games, Automata, Logics and Formal Verification, GandALF 2014*, volume 161 of *EPTCS*, pages 116–130, 2014.

C. S. Calude, S. Jain, B. Khoussainov, W. Li, and F. Stephan.  
**Deciding parity games in quasipolynomial time.**  
In *Theory of Computing, STOC 2017*, pages 252–263. ACM, 2017.
C. Cîrstea, C. Kupke, and D. Pattinson.

**EXPTIME tableaux for the coalgebraic \( \mu \)-calculus.**

*Log. Meth. Comput. Sci.*, 7, 2011.

A. Dawar and E. Grädel.

**The descriptive complexity of parity games.**

In *Computer Science Logic, CSL 2008*, volume 5213 of *LNCS*, pages 354–368. Springer, 2008.

D. Pattinson.

**Expressivity Results in the Modal Logic of Coalgebras.**

PhD thesis, Universität München, 2001.