Will Deflation Lead to Depletion? On Non-Monotone Fixed Point Inductions*

Erich Grädel
Aachen University

Stephan Kreutzer
University of Edinburgh

Abstract

We survey logical formalisms based on inflationary and deflationary fixed points, and compare them to the (more familiar) logics based on least and greatest fixed points.

1. Dictionary

Deflation: *reduction in size, importance, or effectiveness; contraction of economic activity resulting in a decline of prices; the erosion of soil by the wind.*

Depletion: *the exhaustion of a principal substance, especially a natural resource; a reduction in number or quantity so as to endanger the ability to function.*

2. Introduction

Fixed point logics extend a basic logical formalism (like first-order logic, conjunctive queries, or propositional modal logic) by constructors for defining fixed points of relational operators. The most influential fixed point formalisms in computer science are based on least and greatest fixed points of monotone operators.

– The modal μ-calculus L_μ is the extension of propositional modal logic by least and greatest fixed points. In terms of expressive power, it subsumes a variety of modal and temporal logics used in verification, in particular LTL, CTL, CTL*, PDL and also many logics from other areas of computer science. On the other hand, L_μ has a rich theory, and is well-behaved in model-theoretic and algorithmic terms.

– LFP, the extension of first-order logic by least and greatest fixed points, is of crucial importance in finite model theory and descriptive complexity, in particular due to its tight connection to polynomial-time computability.

In finite model theory and, to a lesser extent, in database theory, a number of other fixed point constructs have been studied, allowing the definition of fixed points of operators that are not necessarily monotone. Here, we will focus on inflationary and deflationary fixed point inductions and compare them to least and greatest fixed points. We will also show a number of examples and scenarios in which deflationary fixed points arise in a natural way.

It turns out that IFP, the extension of first-order logic by inflationary and deflationary fixed points, has precisely the same expressive power as LFP. This has been known for some time for finite structures [9], but has been established only recently for the general case [13, 14]. In finite model theory, LFP and IFP have, due to their expressive equivalence, often be used interchangeably. Nevertheless, we argue that least and inflationary fixed points have quite different properties. This becomes particularly apparent in the context of modal logic. Indeed the inflationary modal fixed point logic MIC has far more expressive power and very different algorithmic and structural properties than the modal μ-calculus. Finally, we will discuss appropriate model checking games for inflationary fixed point logics.

3. Greatest and Deflationary Fixed Points

In LFP, greatest fixed points are defined by formulae $[\mathbf{gfp}\, R\overline{x}\,.\,\varphi(R,\overline{x})](\overline{a})$ saying that $\overline{a}$ is contained in the greatest set R satisfying $R = \{\overline{x} : \varphi(R,\overline{x})\}$. To make sure that this set exists, we require that the relation variable R appears only positively in φ. This guarantees that the operator $F_\varphi : R \mapsto \{\overline{x} : \varphi(R,\overline{x})\}$ is monotone on every structure (which means that $R \subseteq R'$ implies $F_\varphi(R) \subseteq F_\varphi(R')$), and it is a classical observation, attributed to Knaster and Tarski, that monotone operators always have a greatest (and a least) fixed point. Moreover, the greatest fixed point can be obtained by an iterative process. Starting with the set of all tuples of appropriate arity in the structure under consideration, we repeatedly apply the operator F_φ to obtain a

*This research has been partially supported by the European Research Training Network "Games and Automata for Synthesis and Validation" (GAMES)

decreasing (possibly transfinite) series of stages which converges to the greatest fixed point. A slightly different variant permits also simultaneous fixed point inductions over several formulae, but it can be shown that this does not provide more expressive power.

Deflationary fixed points, on the other hand, can be built with formulae $\varphi(R, \overline{x})$ that need not be positive in R. Starting with $R = A^k$, we can still define a decreasing sequence of stages by iteratively taking the intersection of the current stage R with $F_\varphi(R)$. This sequence as well must eventually converge to a fixed point (not necessarily of F_φ, but of the associated deflationary operator $R \mapsto R \cap F_\varphi(R)$), which we call the deflationary fixed point of φ. We write

$$[\mathbf{dfp}\, R\overline{x}\, .\, \varphi(R, \overline{x})](\overline{a})$$

to express that $\overline{a}$ is contained in the deflationary fixed point of φ. It is easy to see that for monotone operators, the greatest and the deflationary fixed points coincide. However, non-monotone operators do not necessarily have a greatest fixed point, and if it exists, it need not by identical with the deflationary fixed point. We next discuss some scenarios where deflationary fixed points arise naturally.

3.1. Bisimulation

Let $\mathcal{K} = (V, E, P_1, \ldots, P_m)$ be a transition system with a binary transition relation E and unary predicates P_i. Bisimilarity on $\mathcal{K}$ is the maximal equivalence relation $\sim$ on V such that any two equivalent nodes satisfy the same unary predicates P_i and have edges into the same equivalence classes. To put it differently, $\sim$ is the greatest fixed point of the refinement operator $F : \mathcal{P}(V \times V) \to \mathcal{P}(V \times V)$ with

$$\begin{aligned} F : Z \mapsto \{(u, v) \in V \times V : {\textstyle\bigwedge_{i \le m}} P_i u \leftrightarrow P_i v \\ \wedge \forall u'(Euu' \to \exists v'(Evv' \wedge Zu'v')) \\ \wedge \forall v'(Evv' \to \exists u'(Euu' \wedge Zu'v))\}. \end{aligned}$$

For some applications one is interested to have not only the bisimulation relation $\sim$ but also a linear order on the bisimulation quotient $\mathcal{K}/_\sim$. That is, we want to define a pre-order $\preccurlyeq$ on $\mathcal{K}$ such that $u \sim v$ iff $u \preccurlyeq v$ and $v \preccurlyeq u$. We can again do this via a fixed point construction, by defining a sequence $\preccurlyeq_\alpha$ of pre-orders (where α ranges over ordinals) such that $\preccurlyeq_{\alpha+1}$ refines $\preccurlyeq_\alpha$ and $\preccurlyeq_\lambda$, for limit ordinals λ, is the intersection of the pre-orders $\preccurlyeq_\alpha$ with $\alpha < \lambda$. Let

$$u \preccurlyeq_1 v :\Longleftrightarrow \bigwedge_{i \le m} P_i u \to \Big(P_i v \vee \bigvee_{j < i} (\neg P_j u \wedge P_i v)\Big)$$

(i.e. if the truth values of the P_i at u are lexicographically smaller or equal than those at v), and for any α, let

$$u \sim_\alpha v :\Longleftrightarrow u \preccurlyeq_\alpha v \wedge v \preccurlyeq_\alpha u.$$

To define the refinement, we say that the $\sim_\alpha$-class C *separates* two nodes u and v, if precisely one of the two nodes has an edge into C. Now, let $u \preccurlyeq_{\alpha+1} v$ if, and only if, $u \preccurlyeq_\alpha v$ and there is an edge from v (and hence none from u) into the smallest $\sim_\alpha$-class (wrt. $\preccurlyeq_\alpha$) that separates u from v (if it exists). Since the sequence of the pre-orders $\preccurlyeq_\alpha$ is decreasing, it must indeed reach a fixed point $\preccurlyeq$, and it is not hard to show that the corresponding equivalence relation is precisely the bisimilarity relation $\sim$.

The point that we want to stress here is that $\preccurlyeq$ is a deflationary fixed point of a non-monotone induction. Indeed, the refinement operator on pre-orders is not monotone and does, in general, not have a greatest fixed point.

3.2. Iterated Relativisation

Let $\mathfrak{A}$ be a relational structure and $\varphi(x)$ a specification that should be satisfied by all elements. If this is not the case we can try to throw away all elements of $\mathfrak{A}$ that do not satisfy φ, i.e. to relativize $\mathfrak{A}$ to the substructure $\mathfrak{A}|_\varphi$ induced by $\{a : \mathfrak{A} \models \varphi(a)\}$. Unfortunately, it need not be the case that $\mathfrak{A}|_\varphi \models \forall x \varphi(x)$. Indeed, the removal of some elements may have the effect that others do no longer satisfy φ. But we can of course iterate this relativisation procedure and define a (possibly transfinite) sequence of substructures $\mathfrak{A}^\beta$, with $\mathfrak{A}^0 = \mathfrak{A}$, $\mathfrak{A}^{\beta+1} = \mathfrak{A}^\beta|_\varphi$ and $\mathfrak{A}^\lambda = \bigcap_{\beta<\lambda} \mathfrak{A}^\beta$ for limit ordinals λ. This sequence reaches a fixed point $\mathfrak{A}^\infty$ which satisfies $\forall x \varphi(x)$ — but it may be empty.

This process of iterated relativisation is definable by a fixed point induction in $\mathfrak{A}$. Let $\varphi|_Z$ be the syntactic relativisation of φ to a new set variable Z, obtained by replacing inductively all subformulae $\exists y \alpha$ by $\exists y(Zy \wedge \alpha)$ and $\forall y \alpha$ by $\forall y(Zy \to \alpha)$. Iterated relativisation means repeated application of the operator

$$F : Z \mapsto \{a : \mathfrak{A}|_Z \models \varphi(a)\} = \{a : \mathfrak{A} \models Za \wedge \varphi|_Z(a)\}$$

starting with $Z = A$ (the universe of $\mathfrak{A}$). Note that F is deflationary but not necessarily monotone. Thus, the question whether $\mathfrak{A}^\infty$ is empty or not is one instance of the problem in the title: does deflation lead to depletion?

In logics with inflationary and deflationary fixed points (the universe of) $\mathfrak{A}^\infty$ is uniformly definable in $\mathfrak{A}$ by a formula of form $[\mathbf{dfp}\, Zx\, .\, \varphi|_Z](x)$ (see Sections 4 and 5 for precise definitions).

Question. *Is it also definable using just least and greatest fixed points of monotone operators?*

3.3. Knowledge and Public Announcement

Iterated relativisation has a natural meaning also in epistemic logics, i.e. logics of knowledge. For background we refer to [7]. Basic epistemic logic (for a group A of agents and a set of atomic propositions $\{P_b : b \in B\}$) is just propositional modal logic, interpreted on possible-world models, i.e., Kripke structures $\mathcal{K} = (V, (E_a : a \in A), (P_b : b \in B))$, where each possibility relation E_a is an equivalence relation. The intended meaning of $[a]\varphi$ is "agent a knows φ", which is true in a world $v \in V$ if φ holds in all worlds w that agent a considers possible in world v.

A key concept in epistemic logics is *common knowledge*. A proposition φ is common knowledge at a world v (in short: $\mathcal{K}, v \models C\varphi$) if everybody knows φ, and everybody knows that everybody knows φ, and everybody knows that everybody knows that everybody knows .... Clearly, common knowledge is a greatest fixed point. In the modal μ-calculus, $C\varphi$ is defined by $\nu X.\varphi \wedge \bigwedge_{a\in A}[a]X$.

Suppose now that somebody (who is trusted by all agents) publicly announces φ. One would think that by this action, φ has become common knowledge, since everybody has learned that φ is true and everybody has learned that everybody has learned, and so on. Indeed, the announcement changes the state of knowledge of the agents, and thus induces an update of the model: all worlds which currently do not satisfy φ are eliminated, in other words, $\mathcal{K}$ is relativised to φ. Epistemic logics with public announcement (as considered for instance in [2, 15]) admit formulae $[\varphi!]\psi$ expressing that ψ holds after announcement of φ, i.e., after the model has been relativised to φ. Of course this can easily be captured via syntactic relativisation so it does not go beyond basic epistemic logic (if common knowledge is present, it has to be expanded as a greatest fixed point before relativisation).

However, it is important to note that in the updated model $\mathcal{K}|_\varphi$, φ is *not* necessarily common knowledge. Consider announcements involving ignorance like $\neg[a][b]\psi$ ("a considers it possible that b does not know ψ"). Removal of those worlds where this is false may have the effect that at others, agent a now knows that b knows, so the announced statement becomes false there by its very announcement. But if somebody keeps announcing φ after each relativisation step, we have a process of iterated relativisation that will eventually restrict the model to the deflationary fixed point $(\mathbf{dfp}\, X \leftarrow \varphi|_X)$. We can again ask if this fixed point is definable by monotone inductions, but this time in a more specific scenario.

Question. *Let φ be a formula of basic epistemic logic (with or without common knowledge). Is the iterated relativisation by φ definable in the modal μ-calculus?*

We are grateful to Johan van Benthem for asking this question and for pointing out to us the connection between public announcement and relativisation. We will answer the question in Section 5.

4. Fixed Point Extensions of First-Order Logic

The first systematic studies of least and inflationary fixed points on abstract structures appeared in the 1970s, see [1, 16, 17]. At that time the focus was on monotone and non-monotone inductions over first-order formulae. No explicit fixed point operators were added to the language of first-order logic, fixed points were not being nested, and not interleaved with other logical operations. Despite these differences with the fixed point logics as they are studied today, many methods fundamental to today's theory of fixed point logics originate from the work done at that time.

Fixed point logics in the modern sense appeared independently in several areas of logic in computer science, such as database theory, finite model theory, and verification. Their importance comes from the observation that recursion or unbounded iteration can be modelled elegantly by fixed point constructs. We will briefly recall some basic definitions here. For a more extensive introduction to fixed point extensions of first-order logic, see [6, 8].

A formula $\varphi(R, \overline{x})$ with a free k-ary second-order variable and a free k-tuple of first-order variables $\overline{x}$ defines, on every structure $\mathfrak{A}$, a relational operator $F_\varphi : \mathcal{P}(A^k) \to \mathcal{P}(A^k)$ taking $R \subseteq A^k$ to the set $\{\overline{a} : (\mathfrak{A}, R) \models \varphi(\overline{a})\}$. Fixed point extensions of first-order logic are obtained by adding to FO explicit constructs to form fixed points of definable operators. The type of fixed points that are used determines the expressive power but also the algorithmic complexity of the resulting logics. The most important of these extensions are least fixed point logic (LFP) and inflationary fixed point logic (IFP).

The inflationary fixed point of any operator $F : \mathcal{P}(A^k) \to \mathcal{P}(A^k)$ is defined as the fixed point of the increasing sequence of sets $(R^\alpha)_{\alpha\in\mathrm{Ord}}$ defined as

$$\begin{array}{lll} R^0 & := & \emptyset, \\ R^{\alpha+1} & := & R^\alpha \cup F(R^\alpha), \text{ and} \\ R^\lambda & := & \bigcup_{\alpha<\lambda} R^\alpha \quad \text{for limit ordinals } \lambda. \end{array}$$

The *deflationary fixed point* of F is constructed in the dual way starting with A^k as the initial stage and taking intersections at successor and limit ordinals.

Definition 4.1. Inflationary fixed point logic (IFP) *is obtained from* FO *by allowing formulae of the form* $[\mathbf{ifp}\, R\overline{x} \,.\, \varphi(R, \overline{x})](\overline{x})$ *and* $[\mathbf{dfp}\, R\overline{x} \,.\, \varphi(R, \overline{x})](\overline{x})$, *for arbitrary φ, defining the inflationary and deflationary fixed point of the operator induced by φ.*

Much more popular than inflationary fixed point logics are logics that are based on least and greatest fixed points of monotone operators (i.e., operators that preserve inclusion). Every monotone operator has a least and a greatest fixed point, which can be defined as, respectively, the intersection and the union, of all fixed points, but which can also be constructed by transfinite induction. For the least fixed point, the *stages* are defined by $X^0 := \emptyset$, $X^{\alpha+1} := F(X^\alpha)$, and $X^\lambda := \bigcup_{\alpha<\lambda} X^\alpha$ for limit ordinals λ. By the monotonicity of F, the sequence of stages increases until it reaches the least fixed point.

However, the property of a formula φ to define a monotone operator is undecidable. As a decidable syntax is an essential feature of a logic, one guarantees monotonicity of F_φ by the condition that the fixed point variable R must occur only positive in the formula $\varphi(R, \overline{x})$.

Definition 4.2. Least fixed point logic (LFP) *is obtained from* FO *by allowing formulae* $[\mathbf{lfp}\, R\overline{x} \,.\, \varphi(R, \overline{x})](\overline{x})$ *and* $[\mathbf{gfp}\, R\overline{x} \,.\, \varphi(R, \overline{x})](\overline{x})$*, for* LFP*-formulae* φ *positive in* R*, defining the least and the greatest fixed point of* φ.

The dualities between least and greatest fixed points, and between inflationary and deflationary fixed points, imply

$$[\mathbf{gfp}\, R\overline{x} \,.\, \psi](\overline{t}) \equiv \neg[\mathbf{lfp}\, R\overline{x} \,.\, \neg\psi[R/\neg R]](\overline{t})$$
$$[\mathbf{dfp}\, R\overline{x} \,.\, \psi](\overline{t}) \equiv \neg[\mathbf{ifp}\, R\overline{x} \,.\, \neg\psi[R/\neg R]](\overline{t}).$$

Hence every LFP- or IFP-formula can be brought into *negation normal form*, where negation applies to atoms only.

Clearly, if $\varphi(R, \overline{x})$ is positive in R, then the stages of the least and the inflationary fixed point induction coincide, and so do the fixed points. It follows immediately that LFP is contained in IFP.

Least and inflationary fixed point logic as introduced here are of central concern in finite model theory and descriptive complexity. This is due to the close relationship with computational complexity classes. This correspondence is made precise in the results by Immerman and Vardi [10, 20] who showed that LFP, as well as IFP, characterise polynomial time computability on (ordered) finite structures.

Definition 4.3. *A logic $\mathcal{L}$ captures a complexity class $\mathcal{C}$ on a domain $\mathcal{D}$ of finite structures if the model checking problem for every fixed formula of $\mathcal{L}$ on structures from $\mathcal{D}$ is in the complexity class $\mathcal{C}$, and if every class of structures in $\mathcal{D}$, whose membership problem is in $\mathcal{C}$, is definable on $\mathcal{D}$ by a sentence of $\mathcal{L}$.*

Theorem 4.4 (Immerman, Vardi). LFP *and* IFP *both capture* PTIME *on the class of ordered finite structures.*

Similar results have been established for most major complexity classes using extensions of first-order logic by different fixed point constructs. Capturing results are a central concern of finite model theory; they allow the transfer of methods from logic to complexity theory and vice versa. A major limitation though is the restriction to ordered structures. Most of the known capturing results fail in the absence of a linear order. For instance, it is easily seen that without an ordering LFP falls short of capturing all of PTIME. A typical example of this failure is the class of finite sets of even cardinality, that can of course easily be decided in polynomial time, but is not definable in LFP.

Since both logics capture PTIME, IFP and LFP are equivalent on ordered finite structures. What about unordered structures? It was shown by Gurevich and Shelah [9] that the equivalence of IFP and LFP holds on all finite structures. Their proof does not work on infinite structures, and indeed, there are some important aspects in which least and inflationary inductions behave differently. For instance, there are first-order operators (on arithmetic, for instance) whose inflationary fixed point is not definable as the least fixed point of a first-order operator. Hence it was conjectured by many that IFP might be more powerful than LFP. However, Kreutzer [13] could show recently that IFP is equivalent to LFP on arbitrary structures. Both proofs, by Gurevich-Shelah and Kreutzer, rely on constructions showing that the *stage comparison relations* of inflationary inductions are definable by **lfp**-inductions.

Definition 4.5. *For every monotone or inflationary operator $F : \mathcal{P}(A^k) \to \mathcal{P}(A^k)$, with stages X^α converging to a fixed point X^∞, the F*-rank *of a tuple $\overline{a}$ is $|\overline{a}|_F := \min\{\alpha : \overline{a} \in X^\alpha\}$ if $\overline{a} \in X^\infty$, and $|\overline{a}|_F = \infty$, otherwise. The* stage comparison relations *of F are defined by*

$$\overline{a} \leq_F \overline{b} \quad \textit{iff} \quad |\overline{a}|_F \leq |\overline{b}|_F < \infty$$
$$a \prec_F b \quad \textit{iff} \quad |\overline{a}|_F < |\overline{b}|_F.$$

Given a formula $\varphi(R, \overline{x})$, we write $\leq_\varphi$ and $\prec_\varphi$ for the stage comparison relations of F_φ, and $\leq_\varphi^{\text{inf}}$ and $\prec_\varphi^{\text{inf}}$ for the stage comparison relations of the associated inflationary operator $G_\varphi : R \mapsto R \cup \{\overline{a} : \mathfrak{A} \models \varphi(R, \overline{a})\}$.

Example 4.6. For the formula $\varphi(T, x, y) := Exy \vee \exists z(Exz \wedge Tzy)$ the relation $\prec_\varphi$ on a graph (V, E) is distance comparison:

$$(a, b) \prec_\varphi (c, d) \text{ iff } \mathrm{dist}(a, b) < \mathrm{dist}(c, d).$$

Stage Comparison Theorems deal with the definability of stage comparison relations. For instance, Moschovakis [16] proved that the stage comparison relations $\leq_\varphi$ and $\prec_\varphi$ of any positive first-order formula φ are definable by a simultaneous induction over positive first-order formulae. For the equivalence results on IFP and LFP one needs a Stage Comparison Theorem for IFP-inductions.

We first observe that the stage comparison relations for IFP inductions are easily definable in IFP (see [17]). Indeed, for any formula $\varphi(T, \overline{x})$, the relation $\prec_\varphi^{\text{inf}}$ is defined by the formula

$$[\mathbf{ifp}\, \overline{x} \prec \overline{y} \,.\, \varphi[T\overline{u}/\overline{u} \prec \overline{x}](\overline{x}) \wedge \neg\varphi[T\overline{u}/\overline{u} \prec \overline{x}](\overline{y})](\overline{x}, \overline{y}),$$

where $\varphi[T\overline{u}/\overline{u} \prec \overline{x}](\overline{x})$ denotes the formula obtained from φ by replacing in φ every atom $T\overline{u}$ by the atom $\overline{u} \prec \overline{x}$ defining the set of tuples of rank less than $\overline{x}$.

However, what one needs to show is that the stage comparison relation for IFP-inductions is in fact LFP-definable.

Theorem 4.7 (Inflationary Stage Comparison). *For any formula $\varphi(R, \overline{x})$ in* FO *or* LFP*, the stage comparison relation $\prec_\varphi^{\text{inf}}$ is definable in* LFP*. On finite structures it is even definable in positive* LFP.

We only sketch the main ideas of the proof; for details, see [6, 9] in the case of finite structures and [13, 14] for the more difficult construction in the general case. Consider again the IFP-formula $\psi := [\mathbf{ifp}\, \overline{x} \prec \overline{y} \,.\, \varphi[T\overline{u}/\overline{u} \prec \overline{x}](\overline{x}) \wedge \neg\varphi[T\overline{u}/\overline{u} \prec \overline{x}](\overline{y})](\overline{x}, \overline{y})$ above defining the relation $\prec_\varphi^{\text{inf}}$. As φ is used both positively and negatively in ψ the fixed point relation $\prec$ also occurs both positively and negatively. To turn this into an LFP-definition, purely positive in $\prec$, one has to replace the negative occurrences of $\prec$ by positive definitions. If we are only concerned with finite structures, there is an elegant trick for doing this.

On finite structures, closure ordinals are always finite. Suppose that the inflationary fixed point of $\varphi(T, \overline{x})$ is reached at stage n on $\mathfrak{A}$. For each non-zero ordinal $\alpha \leq n$, we can choose a tuple $\overline{a}$ of rank α and define the stage T^α by $\{\overline{b} : \overline{b} \leq_\varphi^{\text{inf}} \overline{a}\}$ and its complement by $\{\overline{b} : \overline{a} \prec_\varphi^{\text{inf}} \overline{b}\}$. The idea, now, is to simultaneously define the two stage comparison relations $\prec_\varphi^{\text{inf}}$ and $\leq_\varphi^{\text{inf}}$. Suppose the relations have been defined up to some stage α. To define the next stage $\alpha+1$, one can existentially quantify over a tuple $\overline{z}$ of rank α and replace in $\varphi(T, \overline{x})$ every positive occurrence of an atom $T\overline{u}$ by the atom $\overline{u} \leq \overline{z}$, defining the stage T^α, and every negative occurrence of $T\overline{u}$ by $\overline{z} \prec \overline{u}$ defining the complement of T^α. In this way, the next stage of the induction on $\prec$ and $\leq$ can be defined.

If also infinite structures are considered, this trick does no longer work. The problem is that at limit stages λ, there is no tuple $\overline{z}$ of rank exactly λ which could be used to define the next stage. (Recall that T^λ is just the union of all lower stages.) So one has to construct a positive definition for the complement of a stage T^α in a different way. A solution to this is to define a second formula $\vartheta(\prec, \overline{x}, \overline{y})$ which, for any given interpretation of $\prec$ by a stage $\prec^\alpha$, defines again $\prec^\alpha$ from scratch, but by a positive fixed point induction on a new relation Q using $\prec$ only negatively. Once such a formula is defined, we can replace every negative reference to the fixed point variable $\prec$ by the formula ϑ. As ϑ is negative in $\prec$ using it negatively results in a formula positive in $\prec$ and thus in a proper LFP-definition. Essentially, ϑ can be defined similarly to the formula ψ given above, using for every negative reference to the stage comparison relation the outer fixed point variable $\prec$, and for every positive reference the inner variable Q. Strictly speaking, however, such a formula ϑ does not always exist. Still, using this idea one can show that the stage comparison relation $\prec_\varphi^{\text{inf}}$ can be defined in LFP. Note, however, that the stage comparison relation is defined by two nested fixed points the inner of which occurs negatively. We will see below that this cannot be avoided.

From this result, the equivalence of LFP and IFP follows easily.

Theorem 4.8 (Kreutzer). *For every* IFP-*formula there is an equivalent* LFP-*formula.*

Proof. $[\mathbf{ifp}\, R\overline{x} \,.\, \varphi(R, \overline{x})](\overline{x}) \equiv \varphi(\{\overline{y} : \overline{y} \prec_\varphi^{\text{inf}} \overline{x}\}, \overline{x})$. $\square$

Alternation hierarchies for LFP and IFP. Stage comparison theorems have also other interesting consequences. For instance, Moschovakis' Theorem implies that on finite structures, greatest fixed points can be expressed using only least fixed points (that do not occur in the scope of negation). The proof uses, again, that on finite structures there are tuples $\overline{a}$ of maximal rank with respect to $\leq_\varphi^{\text{inf}}$ and for any such $\overline{a}$, the fixed point of φ consists of all $\overline{b} \leq_\varphi^{\text{inf}} \overline{a}$ and its complement of the elements $\overline{b}$ such that $\overline{a} \prec_\varphi^{\text{inf}} \overline{b}$. Since the set of tuples with maximal rank maximal is definable, one can express every greatest fixed point by a least fixed point. This gives a simple normal form for LFP and IFP on finite structures (see [10]).

However, this approach fails on infinite structures. Fixed points may be reached at limit stages and there may be no elements of maximal rank. And indeed, on infinite structures the alternation hierarchy of least and greatest fixed points is strict (see [16] and [3]). For IFP though, the alternation hierarchy collapses, i.e. any deflationary fixed point can be translated to an inflationary fixed point. This is particularly simple if simultaneous inductions over several formulae are permitted (see also Definition 5.1 below). Indeed $\neg[\mathbf{ifp}\, R\overline{x} \,.\, \varphi(R, \overline{x})](\overline{x})$ is equivalent to the formula $[\mathbf{ifp}\, P\overline{x} : S](\overline{x})$, where

$$S := \begin{cases} R\overline{x} & \leftarrow \quad \varphi(R, \overline{x}) \\ P\overline{x} & \leftarrow \quad \forall \overline{y}(R\overline{y} \leftrightarrow \varphi(R, \overline{y})) \wedge \neg R\overline{x}. \end{cases}$$

The induction on S builds up the inflationary fixed point of φ in R and, once this fixed point has been reached, i.e., $\forall \overline{y}(R\overline{y} \leftrightarrow \varphi(R, \overline{y}))$ becomes true, adds to P all tuples not in R. It is known that simultaneous inductions can be translated to simple ones for both LFP and IFP.

Interestingly, if we are not interested in the alternation of least and greatest (inflationary and deflationary) fixed points but in the nesting of fixed points of the same kind, the situation is in some sense reversed: nested least fixed points can always be replaced by a single one (of larger arity), whereas this fails for inflationary fixed points. Indeed, every deflationary fixed point can be translated to an inflationary one. Thus, if nested inflationary fixed points could be reduced to a single fixed point we would prove that every IFP-formula is equivalent to a formula with only one fixed point operator. A simple diagonalisation argument shows that this is impossible.

On the other hand, it is known that every LFP-formula of the form $[\mathbf{lfp}\ R\overline{x} \,.\, \varphi(R, \overline{x})]$ such that φ contains a positive subformula $\vartheta := [\mathbf{lfp}\ Q\overline{y} \,.\, \psi(Q, R, \overline{y})](\overline{y})$ is equivalent to the simultaneous fixed point $[\mathbf{lfp}\, R\overline{x} : S]$ of the system

$$S := \begin{cases} R\overline{x} & \leftarrow \quad \varphi(R, \overline{x}, \vartheta(\overline{u})/Q\overline{u}) \\ Q\overline{y} & \leftarrow \quad \psi(Q, R, \overline{y}) \end{cases}$$

which can easily be expressed by a single least fixed point of larger arity.

In finite model theory, due to the Gurevich-Shelah Theorem, the two logics LFP and IFP have often been used interchangeably. However, there are significant differences that are sometimes overlooked. Despite the equivalence of IFP and LFP, inflationary and deflationary inductions are more powerful concepts than monotone inductions. The translation from IFP-formulae to equivalent LFP-formulae can make the formulae much more complicated, requires an increase of the arity of fixed point variables and, in the case of infinite structures, introduces alternations between least and greatest fixed points. Indeed it is often much more convenient to use inflationary or deflationary inductions in explicit constructions, the advantage being that one is not restricted to inductions over positive formulae. A case in point is the linear order on the bisimulation quotient that we defined in Section 3.1. By the equivalence of IFP and LFP the deflationary induction defining this order can be translated into a definition based on least fixed points only. However, it is not at all clear how to do this directly, or whether it can be done at all without increasing the arity of the fixed point variables.

5. Modal Fixed Point Logics

Given the close relationship between LFP and IFP, and the importance of the μ-calculus, it is natural to study also the properties and expressive power of inflationary fixed points in modal logic. We assume familiarity with propositional modal logic ML and the μ-calculus L_μ. For a Kripke structure $\mathcal{K}$ and a formula φ, we write $[\![\varphi]\!]^{\mathcal{K}}$ for the set of elements of $\mathcal{K}$ at which φ is true. An analogue of IFP for modal logic is the *modal iteration calculus* MIC introduced in [4], which extends basic multi-modal logic by simultaneous inflationary (and deflationary) inductions.

Definition 5.1. *The modal iteration calculus MIC extends propositional modal logic by the following rule: if $\varphi_1, \dots, \varphi_k$ are formulae of MIC, and $X_1, \dots, X_k$ are propositional variables, then*

$$S := \begin{cases} X_1 & \leftarrow & \varphi_1 \\ & \vdots & \\ X_k & \leftarrow & \varphi_k \end{cases}$$

is a system *of rules, and* $(\mathbf{ifp}\, X_i : S)$ *and* $(\mathbf{dfp}\, X_i : S)$ *are formulae of MIC. If S consists of a single rule $X \leftarrow \varphi$ we simplify the notation and write* $(\mathbf{ifp}\, X \leftarrow \varphi)$ *instead of* $(\mathbf{ifp}\, X : X \leftarrow \varphi)$*, and similarly for* $\mathbf{dfp}$.

We just describe the semantics for **ifp**-formulae. On each Kripke structure $\mathcal{K}$, the system S defines, for each ordinal α, a tuple $\overline{X}^\alpha = (X_1^\alpha, \dots, X_k^\alpha)$ of sets of states, via

$$\begin{aligned} X_i^0 &:= \emptyset, \\ X_i^{\alpha+1} &:= X_i^\alpha \cup [\![\varphi_i]\!]^{(\mathcal{K}, \overline{X}^\alpha)}, \\ X_i^\alpha &:= \bigcup_{\beta<\alpha} X_i^\beta \text{ if } \alpha \text{ is a limit ordinal.} \end{aligned}$$

We call $(X_1^\alpha, \dots, X_k^\alpha)$ the stage α of the inflationary induction of S on $\mathcal{K}$. As the stages are increasing (i.e. $X_i^\alpha \subseteq X_i^\beta$ for any $\alpha < \beta$), this induction reaches a fixed point $(X_1^\infty, \dots, X_k^\infty)$. Now we put $[\![(\mathbf{ifp}\, X_i : S)]\!]^{\mathcal{K}} := X_i^\infty$.

It is clear that MIC is a modal logic in the sense that it is invariant under bisimulation. In fact, on every class of bounded cardinality, inflationary fixed points can be unwound to obtain an equivalent infinitary modal formulae. As a consequence, MIC has the tree model property. It is also clear that MIC is at least as expressive as L_μ. Given that the corresponding extensions of first-order logic by least and inflationary fixed points are equivalent, it is natural to ask whether the step from the μ-calculus to a corresponding non-monotone fixed point calculus does indeed produce something new. In particular, we can pose the following questions.

(1) Is MIC more expressive than L_μ?
(2) Does MIC have the finite model property?
(3) What are the algorithmic properties of MIC? Is the satisfiability problem decidable? Can model checking be performed efficiently (as efficiently as for L_μ)?
(4) Can we eliminate, as in the μ-calculus and as in IFP, simultaneous inductions without losing expressive power?
(5) What is the relationship of MIC with monadic second-order logic (MSO) and with finite automata?
(6) Is MIC the bisimulation-invariant fragment of any natural logic?

In [4] it is shown that indeed, the modal iteration calculus MIC has much greater expressive power than L_μ. But greater expressive power comes at a cost: the calculus is algorithmically much less manageable.

Expressive power. Whereas the μ-calculus has the finite model property, we can axiomatise trees of infinite height in MIC. A well-founded tree is a tree satisfying the formula $\mathbf{ifp}\, X \leftarrow \Box X$. The *height* $h(v)$ of a node v in a well-founded tree $\mathcal{T}$ is an ordinal, namely the least strict upper bound of the heights of its children. For any node v in a tree $\mathcal{T}$, we write $\mathcal{T}(v)$ for the subtree of $\mathcal{T}$ with root v. We first show that the nodes of finite height and the nodes of height ω are definable in MIC.

Lemma 5.2. *Let S be the system*

$$\begin{aligned} X &\leftarrow \Box \mathit{false} \lor (\Box X \land \Diamond \neg Y) \\ Y &\leftarrow X. \end{aligned}$$

Then, on every tree $\mathcal{T}$, $[\![\mathbf{ifp}\, X : S]\!]^{\mathcal{T}} = [\![\mathbf{ifp}\, Y : S]\!]^{\mathcal{T}} = \{v : h(v) < \omega\}$.

Proof. By induction we see that for each $i < \omega$, $X^i = \{v : h(v) < i\}$ and $Y^i = X^{i-1} = \{v : h(v) < i-1\}$. As a consequence $X^\omega = Y^\omega = \{v : h(v) < \omega\}$. One further iteration shows that $X^{\omega+1} = Y^{\omega+1} = X^\omega$. $\square$

With the system S exhibited in Lemma 5.2 we obtain the formulae

finite-height := $(\mathbf{ifp}\, X : S)$

ω-height := ¬finite-height ∧ □finite-height

which define, respectively, the nodes of finite height and the nodes of height ω. Note that ω-height is a satisfiable formula all of whose models are infinite.

Proposition 5.3. *The finite model property fails for MIC.*

MIC versus MSO. There is another interesting aspect showing that MIC has more expressive power than L_μ. The modal μ-calculus is contained in monadic second-order logic MSO, in particular, all L_μ-definable languages are regular. On the other side, it is shown in [4] that there are MIC-definable languages that are not even context-free. It follows that MIC is not contained in MSO.

Concerning the expressive power of MIC on finite words, we remark that all languages in DTIME$(O(n))$ are MIC-definable. On the other hand, as MIC has linear space data complexity, all languages definable in MIC are context-sensitive.

A further interesting example separating MIC from MSO is related to the process of iterated relativisation described in Sections 3.2 and 3.3. Let $\varphi := \langle a\rangle \mathit{true} \vee (r \leftrightarrow [b]\mathit{false})$. The iterated relativisation by φ is described by the MIC-formula $\psi := (\mathbf{dfp}\, X \leftarrow \varphi|_X)$.

Lemma 5.4. *No formula in* MSO *is equivalent to* ψ.

Proof. To see this, consider the formula ψ (or equivalently, the process of iterated relativisation defined by φ) on finite trees $T(n,m)$, consisting of a root r with two branches attached: a branch of length n of a-transitions, and a branch of length m of b-transitions. The root r is identified by the atomic proposition that is also denoted r.

It is easy to see that $T(n,m), r \models \psi$ iff $n \geq m$. Indeed after $k \leq \min(m,n)$ relativisation steps, $T(n,m)$ is reduced to $T(n-k, m-k)$. Further, note that $T(n,m), r \models \varphi$ if, and only if, $n > 0$ or $m = 0$.

Hence, if $n < m$, then the first m relativisation steps produce the tree $T(0, m-n)$, and the following step will remove the root. Further iterations will then reduce the structure to the empty set. However, if $n \geq m$, then the first m relativisation steps produce $T(n-m, 0)$, and m further steps will reduce the tree to $T(0,0)$, that is to the tree consisting just of the root, and this is the fixed point.

On the other side, a straightforward pumping arguments shows that no finite tree automaton can accept precisely those trees $T(n,m)$ for which $n \geq m$. Since every MSO-sentence is equivalent, on trees, to a finite automaton, ψ is not equivalent to any MSO-formula. $\square$

Hence, no formula in the modal μ-calculus is equivalent to ψ. It is not difficult to modify this example so that it also works for epistemic logics (i.e. Kripke structures where the accessibility relations are equivalence relations).

Corollary 5.5. *Epistemic logics with iterated public announcement cannot be embedded into the modal μ-calculus.*

Janin and Walukiewicz [11] have shown that every property of transition systems that is invariant under bisimulation and definable in MSO, can actually be defined in L_μ. Thus, the μ-calculus gives a precise characterisation of the bisimulation invariant properties definable in MSO, or, to put it more concisely, L_μ is the bisimulation-invariant fragment of MSO. The question arises whether we can similarly characterise MIC as the bisimulation invariant fragment of some natural extension of MSO.

Given that the monadic fragment M-LFP of LFP coincides with MSO on trees and thus, the bisimulation invariant fragments of M-LFP and MSO fall together, a natural question is whether MIC is the bisimulation invariant fragment of M-IFP, the monadic fragment of IFP. However this was refuted in [5]; it turned out that the bisimulation invariant fragment of M-IFP is far more expressive than MIC.

Another natural candidate is the multi-dimensional μ-calculus introduced by Otto [18], which characterises the bisimulation-invariant fragment of PTIME. However, although MIC is much more expressive than L_μ, it falls short of defining all bisimulation-invariant polynomial-time properties. In particular, as shown in [4], bisimulation itself is not definable in MIC. Hence it is open, whether MIC can be characterised in this way.

Algorithmic properties. The two fundamental algorithmic problems related to a logic are the satisfiability problem and model checking. For the modal μ-calculus, satisfiability is decidable in EXPTIME and model checking is known to be in NP ∩ co-NP (and conjectured by many to be solvable in polynomial time). Unfortunately, MIC has much less convenient algorithmic properties.

Theorem 5.6. *The satisfiability problem for MIC is undecidable. In fact, it is not even in the arithmetical hierarchy.*

To prove this, one can again work with well-founded trees, and show that arithmetic on the heights is definable in MIC. For details, see [4]. Furthermore it has recently been shown by Miller and Moss [15], that even very small fragments of MIC are undecidable, and in fact Σ^1_1-hard. It suffices to add to basic propositional logic the process of iterated relativisation, i.e. a very restricted form of deflationary fixed points.

Theorem 5.7. *The logic of iterated public announcement is undecidable, even without common knowledge.*

The straightforward bottom-up evaluation method shows that model checking for MIC is polynomial-time in the size of the structure and polynomial space in the size of the formula. Unfortunately, this naive approach cannot be improved essentially [4].

Theorem 5.8. *The expression complexity of MIC-model checking is* PSPACE*-complete, even without simultaneous inductions and even on very simple structures with just two elements.*

We finally remark that although simultaneous fixed point inductions in LFP, IFP, or the modal μ-calculus can be eliminated without losing expressive power, this is not the case for MIC. However, all the expressiveness and complexity results for MIC that we mentioned survive if only simple inductions are permitted.

We have seen that in the context of modal logic, inflationary and deflationary fixed points are much stronger than least and greatest fixed points, and MIC has very different structural properties compared to L_μ. However, in terms of algorithmic properties we pay a (high) price for this increased expressive power. Indeed the complexity results that we have shown probably exclude MIC as useful formalism for, say, hardware verification.

6. Games for Inflationary Fixed Points

Model checking problems, for almost any logic, can be reformulated as strategy problems for appropriate model checking games. With a formula ψ and a structure $\mathfrak{A}$ we associate a game $\mathcal{G}(\mathfrak{A}, \psi)$ played by two players, *Verifier* and *Falsifier*. Verifier (sometimes also called Player 0) tries to prove that $\mathfrak{A} \models \psi$, whereas Falsifier (also called Player 1) tries to establish that the formula is false. For first-order logic or propositional modal logic, evaluation games are very simple in the sense that winning conditions are *positional*, and that the games are *well-founded*, i.e. all possible plays are finite (regardless of whether the input structure is finite or infinite).

The appropriate model checking games for LFP and for the modal μ-calculus are parity games. These are infinite games where positions have a *priority*, and the winner of an infinite play is determined according to whether the least priority seen infinitely often during the play is even or odd. It is open whether winning sets and winning strategies for parity games can be computed in polynomial time. The best algorithms known today are polynomial in the size of the game, but exponential with respect to the number of priorities. Competitive model checking algorithms for the modal μ-calculus work by solving the strategy problem for the associated parity game (see, e.g., [12]).

The question arises whether one can generalise parity games to appropriate evaluation games for IFP and MIC. Note that there is a trivial possibility to define a model checking game for IFP, namely to unwind all fixed points and take the associated first-order game. But this is unsatisfactory in several respects. The main problem is the explosion of the game graph: for each fixed point of arity k the number of positions increases by a factor n^k (where n is the size of the input structure). Hence, even if we only have monadic fixed points (as is the case for MIC), the size of the game graph would be exponential in the number of fixed point operators.

We would like instead to define a model checking game with (essentially) the same game graphs as for least fixed point logics, and take care of the increased power by different winning conditions.

Let us recall the definitions of model checking games for ML and the modal μ-calculus (the games for FO and LFP are analogous). Consider a Kripke structure $\mathcal{K} = (V, (E_a : a \in A), (P_b : b \in B))$ and a formula $\psi \in L_\mu$ which we may assume to be in negation normal form and *well-named*, in the sense that every fixed-point variable is bound only once.

The game $\mathcal{G}(\mathcal{K}, \psi)$ is a parity game whose positions are pairs (φ, v) such that φ is a subformula of ψ, and v is a node of $\mathcal{K}$. Player 0 (Verifier) moves at positions associated to disjunctions and formulae $\langle a \rangle \varphi$. From a position $(\varphi \vee \vartheta, v)$ she moves to either (φ, v) or (ϑ, v) and from a position $(\langle a \rangle \varphi, v)$ she can move to any position (φ, w) such that $(v, w) \in E_a$. In addition, Verifier is supposed to move at atomic false positions, i.e., at positions (P_b, v) where $v \notin P_b$ and $(\neg P_b, v)$ where $v \in P_b$. However, these positions do not have successors, so Verifier loses at atomic false positions. Dually, Player 1 (Falsifier) moves at conjunctions and formulae $[a]\varphi$, and loses at atomic true positions. The rules described so far determine the model checking game for ML-formulae ψ and it is easily seen that Verifier has a winning strategy in this game $\mathcal{G}(\mathcal{K}, \psi)$ starting at position (ψ, v) if, and only if, $\mathcal{K}, v \models \psi$.

For formulae in L_μ, we also have positions $(\lambda X.\varphi, v)$ (where $\lambda \in \{\mu, \nu\}$) and (X, v), for fixed-point variables X. At these positions there is a unique move (by Falsifier, say) to (φ, v), i.e. to the formula defining the fixed point. The priority labelling assigns odd priorities to ν-variables and even priorities to μ-variables. Further, if X, X' are fixed-point variables of different kind with X' depending on X (which means that X occurs free in the formula defining X'), then X-positions get lower priority than X'-positions. The remaining positions, not associated with fixed-point variables, do not have a priority (or have the maximal one).

For more details and explanations, and for the proof that the construction is correct, see e.g. [8, 19].

Theorem 6.1. $\mathcal{K}, v \models \psi$ *if, and only if, Verifier has a winning strategy for the parity game* $\mathcal{G}(\mathcal{K}, \psi)$ *from position* (ψ, v).

For LFP an analogous construction works, but the game graph may become much larger, especially if the width of the formulae (the maximal number of free variables in subformulae) is large. For LFP-formulae where both the alternation depth and the width are bounded, the model checking problem can be solved in polynomial time (for instance via solving the model checking game). However, the model checking problem for LFP is EXPTIME-complete for formulae of unbounded width, even if there is only one application of an LFP-operator. The important unresolved case concerns LFP-formulae with bounded width, but unbounded alternation depth. This includes the μ-calculus, since every formula of L_μ can be translated into an equivalent LFP-formula of width two. In fact the following three problems are algorithmically equivalent, in the sense that if one of them admits a polynomial-time algorithm, then all of them do.

(1) Computing winning sets in parity games.

(2) The model checking problem for LFP-formulae of width at most k, for any $k \geq 2$.

(3) The model checking problem for the modal μ-calculus.

A game for MIC. For simplicity we will focus on MIC-formulae with a single fixed point, of form $\psi := (\textbf{ifp}\, X \leftarrow \varphi)$ where φ is in propositional modal logic. We always assume that formulae are in negation normal form, and write $\overline{\vartheta}$ for the negation normal form of $\neg\vartheta$.

Let $\psi := \textbf{ifp}\, X \leftarrow \varphi$, where φ is a formula in ML. In general, φ can have positive or negative occurrences of the fixed point variable X. We use the notation $\varphi(X, \overline{X})$ to separate positive and negative occurrences of X. Fix a finite transition system $\mathcal{K}$. The model checking game $\mathcal{G}(\mathcal{K}, \psi)$ is defined as follows. The positions are pairs (ϑ, v) such that v is a node of $\mathcal{K}$ and ϑ is either ψ or a subformula of $X \vee \varphi$ or $\overline{X} \wedge \overline{\varphi}$. Any position of form (X, v) or $(\overline{X}, v)$ is called an X-position.

The game graph allows for the usual moves in the ML-game, i.e. Verifier moves from a position $(\varphi_1 \vee \varphi_2, u)$ to (φ_1, u) or (φ_2, u) and so forth. In addition, we have regeneration moves and backtrack moves.

Regeneration moves. As in games for L_μ, when a play reaches a fixed point variable, then it proceeds to the formula defining that fixed point. Here this means that from positions (X, w) Verifier proceeds to $(X \vee \varphi, w)$ and from positions $(\overline{X}, w)$ Falsifier proceeds to $(\overline{X} \wedge \overline{\varphi}, w)$.

Backtrack moves. From positions (X, w), instead of regeneration, Verifier can also reset the game to the initial position (ψ, v), and Falsifier can do the same at $(\overline{X}, w)$. This is called a backtrack move. The winning condition ensures that only one backtrack move will occur in a play.

Winning condition.

(1) Infinite plays are won by Falsifier.

(2) Any player loses immediately when she backtracks a second time.

(3) Suppose that after the play has gone through α X-positions, one of the players has backtracked. After that, when α further X-positions $(Y_1, u_1), \ldots, (Y_\alpha, u_\alpha)$ have been visited the play ends. Falsifier has won, if $Y_\alpha = X$, and Verifier has won if $Y_\alpha = \overline{X}$.

(4) Whenever a player cannot move, she loses.

We claim that Verifier has a winning strategy for the game $\mathcal{G}(\mathcal{K}, \psi)$ if $\mathcal{K}, v \models \psi$ and Falsifier has a winning strategy if $\mathcal{K}, v \not\models \psi$. Since these games are determined (which is immediate from general facts on infinite games) the converse assertions hold as well.

To prove our claim, we look at the ML-formulae φ^α defining the stages of the induction. Let $\varphi^0 = \mathit{false}$ and $\varphi^{\alpha+1} = \varphi^\alpha \vee \varphi(\varphi^\alpha, \overline{\varphi}^\alpha)$. On finite structures $\psi \equiv \bigvee_{\alpha<\omega} \varphi^\alpha$. Consider the situation after a backtracking move prior to which β X-positions have been visited and suppose that $\mathcal{K}, v \models \varphi^\beta$. A winning strategy for Verifier in an ML-model checking game $\mathcal{G}(\mathcal{K}, \varphi^\beta)$ (from position (φ^β, v)) translates in the obvious way into a (non-positional) strategy for the game $\mathcal{G}(\mathcal{K}, \psi)$ from position (ψ, v) with the following properties: Any play that is consistent with this strategy will either be winning for Verifier before β X-positions have been seen, or the β-th X-position will be negative.

Similarly, if $\mathcal{K}, v \not\models \varphi^\beta$ then Falsifier has a winning strategy for $\mathcal{G}(\mathcal{K}, \varphi^\beta)$, and this strategy translates into a strategy for the game $\mathcal{G}(\mathcal{K}, \psi)$ by which Falsifier forces the play (after backtracking) from position (ψ, v) to a positive β-th X-position, unless she wins before β X-positions have been seen.

Lemma 6.2. *Suppose that a play on* $\mathcal{G}(\mathcal{K}, \psi)$ *has been backtracked to the initial position* (ψ, v) *after* β X*-positions have been visited. Verifier has a winning strategy for the remaining game if, and only if,* $\mathcal{K}, v \models \varphi^\beta$ *and Falsifier has a strategy to win the remaining game if, and only if,* $\mathcal{K}, v \not\models \varphi^\beta$.

From this we obtain the desired result.

Theorem 6.3. *If* $\mathcal{K}, v \models \psi$*, then Verifier wins the game* $\mathcal{G}(\mathcal{K}, \psi)$ *from position* (ψ, v)*. If* $\mathcal{K}, v \not\models \psi$*, then Falsifier wins the game* $\mathcal{G}(\mathcal{K}, \psi)$ *from position* (ψ, v).

Proof. Suppose first that $\mathcal{K}, v \models \psi$. Then there is some ordinal $\alpha < \omega$ such that $\mathcal{K}, v \models \varphi^\alpha$. We construct a winning strategy for Verifier in the game $\mathcal{G}(\mathcal{K}, \psi)$ starting at position (ψ, v).

From (ψ, v) the game proceeds to $(X \vee \varphi, v)$. At this position, Verifier chooses the node (X, v) until this node has been visited α-times. After that, she backtracks and moves to (φ, v). By Lemma 6.2, since $\mathcal{K}, v \models \varphi^\alpha$, Verifier has a strategy to win the remaining play.

Now suppose that $\mathcal{K}, v \not\models \psi$. If, after α X-positions, one of the players backtracks, then Falsifier has a winning strategy for the remaining game, since $\mathcal{K}, v \not\models \varphi^\alpha$. Hence, the only possibility for Verifier to win the game in a finite number of moves is to avoid positions $(\overline{X}, w)$ where Falsifier can backtrack.

Consider the formulae φ_f^α, with $\varphi_f^0 = \mathit{false}$ and $\varphi_f^{\alpha+1} = \varphi(\varphi_f^\alpha, \mathit{false})$. They define the stages of $\mathbf{ifp}\, X \leftarrow \varphi(X, \mathit{false})$, which is obtained from ψ by replacing negative occurrences of X by *false*. If Verifier could force a finite winning play, with $\alpha - 1$ positions of form (X, w) and without positions $(\overline{X}, v)$, then she would in fact have a winning strategy for the model checking game $\mathcal{G}(\mathcal{K}, \varphi_f^\alpha)$. Since ψ_f^α implies φ^α, it would follow that $\mathcal{K}, v \models \varphi^\alpha$. But this is impossible since $\mathcal{K}, v \not\models \psi$. ☐

There are several possible generalisations to a class of games that are powerful enough for arbitrary inflationary fixed points. Here is one possibility.

The set-up is essentially the same as for a parity games. We assume that the priority labelling $\Omega : V \to \mathbb{N}$ is partial (i.e. not all nodes have a priority), but so that on any infinite play there are infinitely many nodes in the range of Ω. In addition to the usual moves along edges there are backtrack moves:

From any node $v \in V_\sigma$ of priority p, Player σ can backtrack to a node u of the same priority provided that

- u has already be seen in the play,
- between u and v, no node of priority $< p$ has been played,
- since the last node of smaller priority, there has not yet been a backtrack move from a node of priority p.

Suppose that a player has backtracked from a node v of priority p, and that α is the number of nodes of priority p between the last node of smaller priority (or the beginning) and v. If, after the backtracking, it happens that α positions $u_1, \dots, u_\alpha$ of priority p, and no node of priority $< p$ are visited, then the player whose turn it is at node u_α loses immediately.

Finally the winner of infinite plays is determined by the parity condition.

It is not too difficult to show that model checking problems for arbitrary formulae from MIC and IFP can indeed be translated to games of this form. However, we have to defer a detailed analysis of the logical and algorithmic properties of this kind of games to a subsequent paper.

References

[1] P. Aczel. An introduction to inductive definitions. In J. Barwise, editor, *Handbook of Mathematical Logic*, volume 90 of *Studies in Logic*, pages 739 –782. North-Holland, 1977.

[2] J. van Benthem. One is a lonely number. Technical Report ILLC Research Report PP-2003-07, Institute for Logic, Language and Computation, University of Amsterdam, 2003.

[3] J. Bradfield. The modal μ-calculus alternation hierarchy is strict. *Theoretical Computer Science*, 195:133–153, 1998.

[4] A. Dawar, E. Grädel, and S. Kreutzer. Inflationary fixed points in modal logic. *ACM Transactions on Computational Logic*, 2003. To appear.

[5] A. Dawar and S. Kreutzer. Generalising automaticity to modal properties of finite structures. In *Proc. 22nd FSTTCS*, LNCS 2556, pages 109–120. Springer, 2002.

[6] H.-D. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer, 2nd edition edition, 1999.

[7] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning About Knowledge*. MIT Press, 1995.

[8] E. Grädel. Finite model theory and descriptive complexity. In *Finite Model Theory and Its Applications*. Springer-Verlag, 2003. To appear.

[9] Y. Gurevich and S. Shelah. Fixed-point extensions of first-order logic. *Annals of Pure and Applied Logic*, 32:265–280, 1986.

[10] N. Immerman. Relational queries computable in polynomial time. *Information and Control*, 68:86–104, 1986.

[11] D. Janin and I. Walukiewicz. On the expressive completeness of the propositional mu-calculus with respect to monadic second order logic. In *Proc. of 7th International Conference on Concurrency Theory CONCUR '96*, LNCS 1119, pages 263–277. Springer-Verlag, 1996.

[12] M. Jurdziński. Small progress measures for solving parity games. In *STACS 2000*, volume 1770 of *LNCS*, pages 290–301. Springer, 2000.

[13] S. Kreutzer. Expressive equivalence of least and inflationary fixed point logic. In *Proceedings of 17th IEEE Symp. on Logic in Computer Science LICS02*, pages 403–410, 2002.

[14] S. Kreutzer. *Pure and Applied Fixed Point Logic*. PhD thesis, Aachen University, 2002.

[15] J. Miller and L. Moss. The undecidability of iterated modal relativization. Manuscript, Indiana University, 2003.

[16] Y. Moschovakis. *Elementary induction on abstract structures*. North Holland, 1974.

[17] Y. Moschovakis. On non-monotone inductive definability. *Fundamentae Mathematica*, 82(82):39–83, 1974.

[18] M. Otto. Bisimulation-invariant Ptime and higher-dimensional mu-calculus. *Theoretical Computer Science*, 224:237–265, 1999.

[19] C. Stirling. Bisimulation, model checking and other games. Notes for the Mathfit instructional meeting on games and computation. Edinburgh, 1997.

[20] M. Vardi. The complexity of relational query languages. In *Proceedings of the 14th ACM Symposium on the Theory of Computing*, pages 137–146, 1982.