Well-founded Semantics and the Algebraic Theory of Non-Monotone Inductive Definitions ⋆

Marc Denecker and Joost Vennekens

Department of Computer Science, Katholieke Universiteit Leuven, Belgium
{marcd,joost}@cs.kuleuven.be

Abstract. Approximation theory is a fixpoint theory of general (monotone and non-monotone) operators which generalizes all main semantics of logic programming, default logic and autoepistemic logic. In this paper, we study inductive constructions using operators and show their confluence to the well-founded fixpoint of the operator. This result is one argument for the thesis that Approximation theory is the fixpoint theory of certain generalised forms of (non-monotone) induction. We also use the result to derive a new, more intuitive definition of the well-founded semantics of logic programs and the semantics of ID-logic, which moreover is easier to implement in model generators.

1 Introduction

This paper studies inductive constructions in relation to the well-founded semantics. The study of induction can be defined as the investigation of a class of effective construction techniques in mathematics. There, sets are frequently defined through a constructive process of iterating some recursive *recipe* that adds new elements to the set given that one has established the presence or absence of other elements in the set. In an inductive definition, this recipe is often represented as a collection of informal rules representing base cases and inductive cases. Inductive rules may be *monotone* or *non-monotone*. Consider for example the well-known definition of satisfiability, denoted $I \models \varphi$, by induction on the structure of (propositional) formulas:

- $I \models P$ if $P \in I$ and P is a propositional variable;
- $I \models \psi \wedge \phi$ if $I \models \psi$ and $I \models \phi$;
- $I \models \neg\psi$ if $I \not\models \psi$.

The third rule states that I satisfies $\neg\psi$ if I does *not* satisfy ψ. This is a non-monotone rule, in the sense that it adds a pair $(I, \neg\psi)$ in absence of the pair (I, ψ), and therefore, applying the "recipe" to sets of formulas does not preserve the order $\subseteq$.

Different forms of inductive constructions have been studied extensively in mathematical logic. Monotone induction was studied starting with [19], and

⋆ Works supported by IWT VLaanderen, FWO Vlaanderen.

later in landmark studies such as [20, 16, 1]. Also non-monotone forms of induction have been studied, such as *inflationary induction* [17] and iterated inductive definitions [10]. In computational logic, the results of these studies were used in extensions of logic with fixpoints constructs, such as FO(LFP) (see, e.g., [9]), μ-calculus and description logics. Inductive definitions are also related to logic programming. It was argued in [3, 8] that the well-founded semantics of logic programming [23] correctly formalizes the semantics of different types of definitions that can be found in mathematics, e.g., recursion-free definitions, monotone inductive definitions, and non-monotone inductive definitions such as inductive definitions over well-founded orders (e.g., the definition of $\models$) and iterated inductive definitions. The fact that the inability to express inductive definitions is a well-known weakness of first order logic, has subsequently motivated an extension of FO with a new construct for representing definitions, whose semantics is based on the well-founded semantics [2, 7, 8]. This logic FO(ID), also called ID-logic, is in some sense an integration of classical logic and logic programming, and can be viewed equally well as a new member of the family of fixpoint logics and even as new (very general sort of) description logic. FO(ID) has recently been proposed as (one of) the underlying language for a constraint programming framework [15].

The study of inductive definitions is strongly related to fixpoint theory. An inductive definition corresponds to an algebraic lattice operator, and, in the monotone case, the object "defined" by such an operator is the least fixpoint. In this sense, Tarski's fixpoint theory of monotone operators [21] can be considered as an entirely abstract algebraic theory of monotone induction. This naturally raises the question whether this theory can be extended to general (monotone and non-monotone) operators in a way that matches with different forms of non-monotone induction. Such extensions are well-known for inflationary induction, but not for induction over a well-founded order or its generalization, iterated induction. But there is a promising candidate. Building on Fitting's work [12] on semantics of logic programming in bilattices, Denecker, Marek and Truszczynski showed that all main types of semantics of a logic program can be characterized algebraically in terms of the three-valued immediate consequence operator of logic programs[4, 6]. The underlying theory, in [4] called *Approximation theory*, is an algebraic fixpoint theory for (bilattice extensions of) general lattice operators which defines the so-called *Kripke-Kleene, stable* and *well-founded* fixpoints of an operator. In case of the immediate consequence operator, these fixpoints are the models of the logic program in the corresponding semantics. This suggests that the well-founded fixpoint construction in this theory is the missing fixpoint theory of iterated induction.

The main goal of this paper is to explore the link between inductive definitions and Approximation theory. As a point of departure, we observe that an inductive definition defines a set by describing how to construct it, and as such, it is essentially a description of a construction process. Informally, such a process starts from the empty set, and proceeds by iteratively applying rules of the definition with a satisfied antecedent, until saturation follows. Of course, as a specification of such a process, an inductive definition is highly non-deterministic. Indeed, in the case of monotone induction, at any intermediate stage, many rules may be applicable. It is a key property of this sort of induction that the order in which the rules are applied does not matter: all such construction processes produce the same outcome. For non-monotone induction, on the other hand, the provided well-founded order must be respected (e.g., applying the third rule to derive $I \models \neg\phi$ can only be done *after* it has been established whether $I \models \phi$), but still, there remain infinitely many ways in which $\models$ can be constructed. The first goal of this paper is to formalize these construction processes and to study how they relate to the well-founded fixpoint construction of Approximation theory. We will define the notion of a *well-founded induction* of a (bilattice extension of an) operator, and demonstrate that all such inductions converge to the well-founded fixpoint.

Secondly, we will concretize this notion of a well-founded induction in the context of ID-logic. This leads to a new, intuitive and much simpler characterization of the well-founded model of a definition (and of a logic program) which does not involve fixpoint operators anymore. Our results thus allow to simplify the definition of the semantics of FO(ID), and of the well-founded semantics of logic programming.

Thirdly, our study of induction sequences has also computational relevance. First, the non-deterministic inference processes that we describe here generalize various methods of well-founded model computation as presented in e.g. [13]. Second, in ASP systems such as SModels [18], and in the FO(ID) model generator MIDL [14], the well-founded model is computed through a kind of constraint propagation mechanism. These systems do not iterate the immediate consequence operator, for this is too expensive. Instead, they iteratively perform inference steps, inferring truth or falsity of an atom with a true, respectively false body, and inferring falsity of unfounded sets. These computation steps are exactly the atomic inference steps that make up a well-founded induction. Thus, our study can give insight in the properties of intermediate objects constructed during such a constraint propagation proces, and may lead to easier correctness proofs for such systems.

2 Approximation theory

Our presentation of Approximation theory is based on [4, 6].

A structure $\langle L, \leq \rangle$ is a poset if $\leq$ is a partial order on the set L, i.e., a reflexive, asymmetric, transitive relation. The relation $\leq$ is a total order if, in addition, for each $x, y \in L, x \leq y$ or $y \leq x$. A subset S of a poset L is a chain if $\leq$ is a total order in S. The structure $\langle L, \leq \rangle$ is chain-complete if each of its chains C has a least upperbound $lub_{\leq}(C)$, and is a complete lattice if each subset $S \subseteq L$ has a least upperbound $lub_{\leq}(S)$ and a greatest lowerbound $glb_{\leq}(S)$. A chain complete poset has a least element $\bot$ and a complete lattice has both a least element $\bot$ and a greatest element $\top$.

Given a poset $\langle L, \leq \rangle$, an operator $O : L \rightarrow L$ is called $\leq$-monotone if O preserves $\leq$, i.e., $x \leq y$ implies $O(x) \leq O(y)$. An element $x \in L$ is a pre-fixpoint of O if $O(x) \leq x$, a fixpoint if $O(x) = x$ and a post-fixpoint if $x \leq O(x)$. The sets of all such $x \in L$ are denoted $Pre(O)$, respectively $Fix(O)$ and $Post(O)$. A monotone operator in a chain complete poset or a complete lattice has a least fixpoint which is also its least pre-fixpoint and the limit of the increasing sequence $\langle x_\xi \rangle_{\xi \geq 0}$, defined by transfinite induction:

- $x_0 = \bot$;
- $x_{\xi+1} = O(x_\xi)$;
- $x_\lambda = lub(\{x_\xi | \xi < \lambda\})$, for limit ordinals λ.

In Approximation theory, pairs $(x, y) \in L^2$ are used to approximate certain elements of L, namely those in the (possibly empty) interval $[x, y] = \{z \in L \mid x \leq z \leq y\}$. Abusing this correspondence between pairs and intervals, we sometimes write $z \in (x, y)$ instead of $z \in [x, y]$, to denote that (x, y) approximates z. On L^2, two natural orders can be defined:

$$\begin{array}{lll} \text{the product order: } & (x, y) \leq (u, v) & \text{if } x \leq u \text{ and } y \leq v \\ \text{the precision order: } & (x, y) \leq_{\mathrm{p}} (u, v) & \text{if } x \leq u \text{ and } v \leq y \end{array}$$

The precision order is the most important. Indeed, if $(x, y) \leq_{\mathrm{p}} (u, v)$, then $[x, y] \supseteq [u, v]$, i.e., (u, v) approximates fewer elements than (x, y). If L is a complete lattice, then both these orders are complete lattice orders in L^2.

In this paper, the relevant pairs of L^2 are the *consistent* pairs. A pair $(x, y) \in L^2$ is consistent if $x \leq y$ (or $[x, y] \neq \emptyset$), and the subset of L^2 consisting of such pairs is denoted L^c. The order $\leq$ is a complete lattice order in L^c, but $\leq_{\mathrm{p}}$ is not, because L^c has no most precise element. However, $\langle L^c, \leq_{\mathrm{p}} \rangle$ is chain-complete. Elements (x, x) of L^2 are called *exact*. The set of exact elements forms a natural embedding of L in L^2. They are also the maximally precise elements of L^c.

Approximation theory studies fixpoints of lattice operators $O : L \to L$ through the use of approximations of O. We define that an operator $A : L^2 \to L^2$ is a *approximator* if it is $\leq_{\mathrm{p}}$-monotone. An approximator is *consistent* if it maps consistent pairs to consistent pairs. An approximator A *approximates* an operator $O \colon L \to L$ (*is an approximation of* O) if for each $x \in L$, $O(x) \in A(x, x)$. Such an operator A provides approximate information on O. Indeed, when $z \in (x, y)$, then $\leq_{\mathrm{p}}$-monotonicity gives us $O(z) \in A(z, z) \subseteq A(x, y)$. Or, $O(z)$ is approximated by $A(x, y)$, and, abusing the duality of pairs and intervals, $O([x, y]) \subseteq A(x, y)$. It is easy to see that when A approximates an operator O, then A is consistent. For this reason, below we only consider consistent approximators. An approximator A is exact if it preserves exactness, i.e., if for all $x \in L$, $A(x, x)$ is exact. In general, an approximator A approximates a collection of lattice operators O, but when A is exact, then the only approximated operator is the operator which maps each $x \in L$ to $A(x, x)_1$ $(= A(x, x)_2)$. An approximator A is *symmetric* if for all $(x, y) \in L^2$, if $A(x, y) = (x', y')$ then $A(y, x) = (y', x')$. A symmetric approximator is exact.

For an approximator A on L^2 and lattice elements $x, y \in L$, the operators $\lambda z.A(z, y)_1$ and $\lambda z.A(x, z)_2$ on L will be denoted $A(\cdot, y)_1$, respectively $A(x, \cdot)_2$. These operators are monotone. We define an operator $(\cdot)^{A\downarrow}$ on L, called the *downward revision operator* of A, as $y^{A\downarrow} = lfp(A(\cdot, y))_1$ for each $y \in L$. We also define the *upward revision operator* $(\cdot)^{A\uparrow}$ of A as $x^{A\uparrow} = lfp(A(x, \cdot)_2)$ for every $x \in L$. Note that if A is symmetric, both operators are identical. We define the *stable operator* $\mathcal{S}t_A : L^2 \to L^2$ of A by $\mathcal{S}t_A(x, y) = (y^{A\downarrow}, x^{A\uparrow})$. It can easily be seen that both $(\cdot)^{A\downarrow}$ and $(\cdot)^{A\uparrow}$ are anti-monotone. It follows that $\mathcal{S}t_A$ is $\leq_p$-monotone.

An approximator A defines a number of different fixpoints: the $\leq_p$-least fixpoint of A, denoted $k(A)$, is called its *Kripke-Kleene fixpoint*, fixpoints of its stable operator $\mathcal{S}t_A$ are *stable fixpoints* and the $\leq_p$-least fixpoint of $\mathcal{S}t_A$, denoted $w(A)$, is called the *well-founded fixpoint* of A. In [4, 5], it was shown that all main semantics of logic programming, autoepistemic logic and default logic can be characterized in terms of the different types of fixpoints of approximation operators associated to theories in these logics. For example, in the context of logic programming, the four-valued van Emden-Kowalski operator $\mathcal{T}_P$ of a logic program P is a symmetric approximation of the two-valued van Emden-Kowalski operator. The downward revision operator of $\mathcal{T}_P$ (which is equal to the upward one, since $\mathcal{T}_P$ is symmetric) coincides with the Gelfond-Lifschitz stable operator P. The Kripke-Kleene, well-founded, stable and exact stable fixpoints of $\mathcal{T}_P$ coincide with, respectively, the Kripke-Kleene model, the well-founded model, the four-valued stable models and the stable models of the logic program P.

Given an approximator A on L^2, we denote by A^c its restriction to L^c. Conversely, any approximator A on L^c, i.e., a $\leq_p$-monotone L^c-operator, can be extended to an approximator on L^2, in many ways. When A is exact then A can be extended to a symmetric approximator on L^2, in many ways. It was shown in [6], that all symmetric extensions of A have the same consistent stable fixpoints, the same well-founded fixpoint and the same exact stable fixpoints. This suggests that consistent stable fixpoints can also be algebraically characterized in terms of A^c. As shown in [6], this is indeed the case but the alternative characterization is slightly more more tedious, mainly because the revision operators $(\cdot)^{A\uparrow}$ and $(\cdot)^{A\downarrow}$ are only partial functions, since $A(\cdot, y)_1$ and $A(x, \cdot)_2$ are not operators on L, but only functions from $[\bot, y]$, respectively $[x, \top]$, to L. Consequently, they may have no least fixpoint.

A lattice operator O can have multiple approximations. This raises the question of how the different types of fixpoints of these approximators relate to each other. By point-wise extension of the precision order $\leq_p$ on L^c, we obtain a precision order between L^c-approximators. When $A \leq_p B$, then any operator O approximated by B is also approximated by A and $k(A) \leq_p k(B)$, $w(A) \leq_p w(B)$, and the set of exact stable fixpoints of A is a subset of that of B. Also, a lattice operator O has a most precise L^c-approximator, called the ultimate approximation. This operator, denoted $\mathcal{U}_O$, maps any tuple (x, y) to $(glb(O([x, y])), lub(O([x, y])))$. Because it is the most precise, its Kripke-Kleene and well-founded fixpoints are the most precise of all approximations of O, and the set of its exact stable fixpoints comprises the exact stable fixpoints of all approximations of O.

The precision order can be further extended to L^2-approximators, by defining $A \leq_{\mathrm{p}} B$ if $A^c \leq_{\mathrm{p}} B^c$ (or, equivalently, $A(x,y) \leq_{\mathrm{p}} B(x,y)$, for each $(x,y) \in L^c$).

3 Monotone and well-founded inductions

Let $\langle L, \leq \rangle$ be a complete lattice and O a monotone operator on L.

Definition 1. *A monotone induction of O is a (possibly transfinite) sequence $\langle x_\xi \rangle_{\xi \leq \alpha}$ such that*

- $x_0 = \bot$;
- $x_\xi < x_{\xi+1} \leq O(x_\xi)$, *for every* $\xi < \alpha$;
- $x_\lambda = lub(\{x_\xi | \xi < \lambda\})$, *for every limit ordinal* $\lambda \leq \alpha$.

A monotone induction $\langle x_\xi \rangle_{\xi \leq \alpha}$ is terminal if it cannot be extended, i.e., there is no $x_{\alpha+1}$ such that $\langle x_\xi \rangle_{\xi \leq \alpha+1}$ is a monotone induction.

Clearly, a monotone induction is an increasing sequence and x_α is its limit. Note that the standard construction of the least fixpoint $lfp(O)$ is a terminal monotone induction. All terminal monotone inductions are confluent, i.e., have the same limit, namely $lfp(O)$.

Proposition 1. *The limit of each terminal monotone induction of O is $lfp(O)$.*

There are many ways in which a set, defined by monotone induction, can be constructed. E.g., the transitive closure T of a graph R can be constructed by an arbitrary process of (non-deterministically) selecting an edge (a,b) from R and adding it to T, or finding a pair $(a,b),(b,c)$ of edges in the current set T and extending this set with (a,c). All these processes lead to the same outcome, namely the transitive closure of R. Proposition 1 formalizes this property.

Let us now investigate the case of arbitrary lattice-operators O. Assume that we have an approximation A of O on L^2. First, note that A is a $\leq_{\mathrm{p}}$-monotone operator, so we can construct monotone inductions with A. Each terminal monotone induction of A constructs the Kripke-Kleene fixpoint $k(A)$.

Observe that a monotone induction of a consistent approximator A consists only of consistent pairs. Therefore, a monotone induction of such an A is also a monotone induction of any more precise operator B, because for any $x_{\xi+1}$ in such a sequence, $x_{\xi+1} \leq_p A(x_\xi)$ then implies that also $x_{\xi+1} \leq_p B(x_\xi)$. It follows from this that $k(A) \leq_{\mathrm{p}} k(B)$, as claimed earlier.

The weakness of the Kripke-Kleene fixpoint construction surfaces when we consider the case that O is monotone. Since $k(A)$ approximates all fixpoints of O, we have $k(A) \leq_{\mathrm{p}} (lfp(O), gfp(O))$. We therefore need to consider more precise constructions.

We call a pair $(x', y') \in L^2$ an A-refinement of $(x,y) \in L^2$ if:

- $(x,y) <_p (x',y') \leq_{\mathrm{p}} A(x,y)$; or
- $x' = x$ and $y' < y$ and $A(x,y')_2 \leq y'$.

Note that the second case is equivalent to saying that y' must be a pre-fixpoint of $A(x, \cdot)_2$. It follows that if $x^{A\uparrow} < y$, then taking $y' = x^{A\uparrow}$ gives us the least value for which (x, y') is an A-refinement by the second rule.

Definition 2. *A* well-founded induction *of A in (x, y) is a sequence $\langle (x_\xi, y_\xi) \rangle_{\xi \leq \alpha}$ such that*

- $(x_0, y_0) = (\bot, \top)$;
- $(x_{\xi+1}, y_{\xi+1})$ *is an A-refinement of (x_ξ, y_ξ), for each $\xi < \alpha$;*
- $(x_\lambda, y_\lambda) = lub(\{(x_\xi, y_\xi) \colon \xi < \lambda\})$*, for limit ordinal $\lambda \leq \alpha$.*

A well-founded induction is terminal *if its limit (x_α, y_α) has no A-refinement.*

A well-founded induction is a $\leq_p$-increasing sequence of pairs with limit (x_α, y_α). The main task now is to prove that well-founded inductions are confluent and produce the well-founded fixpoint. This is the main technical contribution of this paper.

The proof of the convergence of all well-founded inductions is based on an invariance analysis. We will show that all pairs constructed during a well-founded induction satisfy certain invariants and that there is exactly one pair that satisfies these invariants and has no A-refinement. Hence, all well-founded inductions must converge to this pair.

The first invariant is A-contractingness. Recall that all elements in a monotone induction are post-fixpoints. A post-fixpoint (a, b) of A has the interesting property that $O([a, b]) \subseteq A(a, b) \subseteq (a, b)$. Therefore, the operator O is internal in $[a, b]$. In fact, it is *contracting* in $[a, b]$ since $(a, b) \supseteq A(a, b) \supseteq A^2(a, b) \supseteq \ldots$. This property is our motivation for calling a post-fixpoint of A an *A-contracting pair*[1].

Proposition 2. *Each pair in a well-founded induction of A is A-contracting.*

The second invariant aims to express that the lower bound of a pair in an well-founded induction cannot grow too large. For example, if O is monotone, then the pair $(gfp(O), \top)$ could be contracting w.r.t. some approximation A. Unless $lfp(O) = gfp(O)$, this pair would never occur during a well-founded induction because $gfp(O)$ is too large.

Definition 3. *A pair (a, b) is A-prudent if $a \leq x$ for every $x \in L$ such that $A(x, b)_1 \leq x$.*

Equivalently, (a, b) is A-prudent if a is less than each pre-fixpoint x of $A(\cdot, b)_1$, or, more compactly, if $a \leq b^{A\downarrow}$. This definition extends the notion of A-prudence of L^c-approximators in [6] to the case of L^2-approximators.

When O is a monotone operator, then for each symmetric ultimate approximation $\mathcal{U}_O$ of O on L^2, for every pair (x, y), $\mathcal{U}_O(x, y)_1 = O(x)$. Consequently, a pair (a, b) is $\mathcal{U}_O$-prudent if a is less than each pre-fixpoint of O or equivalently, if $a \leq lfp(O)$.

[1] In [6], A-contracting pairs were called A-reliable.

Clearly, the least precise pair $(\bot, \top)$ is A-prudent. Since taking A-refinements and taking limits of A-prudent sequences both preserve A-prudence, we obtain a second invariant.

Proposition 3. *Each pair in a well-founded induction of A is A-prudent.*

The third invariant is consistency. To obtain this, however, we need to impose an additional condition on A.

Definition 4. *We say that an approximator A* gracefully degrades *if for all $(x, y) \in L^2$, $A(y, x)_1 \leq A(x, y)_2$.*

The intuition behind this definition is that the behaviour of such an operator on inconsistent pairs is constrained by its behaviour on consistent pairs. It cannot, for example, map all inconsistent pairs to the most precise pair $(\top, \bot)$. Clearly, a symmetric approximator gracefully degrades.

Lemma 1. *Assume that A degrades gracefully. If (a, b) is A-prudent and consistent, then $a \leq a^{A\uparrow}$.*

Proposition 4. *Each pair in a well-founded induction of a gracefully degrading approximator A is consistent.*

As mentioned in Section 2, all symmetric approximators extending an exact L^c-approximator A have the same consistent stable fixpoints. A more general condition that guarantees this is graceful degradation.

Corollary 1. *Two gracefully degrading L^2-approximators A, B for which $A^c = B^c$, have the same consistent stable fixpoints (and hence, $w(A) = w(B)$).*

A fourth invariant is that each element in a well-founded induction is less than each stable fixpoint. Recall that a stable fixpoint (c, d) satisfies $c = d^{A\downarrow}$ and $d = c^{A\uparrow}$.

Proposition 5. *Let (c, d) be a stable fixpoint of A. If $(a, b) \leq_{\mathrm{p}} (c, d)$, then for each (u, v) such that $(a, b) <_p (u, v) \leq_{\mathrm{p}} A(a, b)$, $(u, v) \leq_{\mathrm{p}} (c, d)$. If $(a, b) \leq_{\mathrm{p}} (c, d)$ then for each $y < b$ such that $A(a, y)_2 \leq y$, $(a, y) \leq_{\mathrm{p}} (c, d)$.*

Clearly, $(\bot, \top)$ approximates all stable fixpoints of A. This property is preserved by taking A-refinements and by taking limits of sequences of increasing precision. From this, we obtain the fourth invariant of well-founded inductions.

Proposition 6. *For each pair (x, y) in a well-founded induction of A and each stable fixpoint of (c, d) of A, $(x, y) \leq_p (x, d)$.*

We have now identified four main invariants. It follows that the limit (x, y) of a well-founded induction is contracting, prudent, less precise than each stable fixpoint of A and, if A gracefully degrades, consistent. In addition, we know that (x, y) has no A-refinement. What can be concluded from this?

Proposition 7. *Let (a, b) be an A-contracting, A-prudent pair such that (a, b) has no A-refinement. Then (a, b) is a stable fixpoint of A.*

Theorem 1. *There exists a least precise stable fixpoint of A, and it is the limit of each terminal well-founded induction of A. If A is gracefully degrading, then this least precise stable fixpoint is consistent.*

This theorem shows that all terminal well-founded inductions indeed reach the same limit and, moreover, that this limit is precisely the well-founded model.

Proposition 8. *Let A, B be gracefully degrading approximators on L^2 such that $A \leq_{\mathrm{p}} B$. A well-founded induction of A is a well-founded induction of B.*

In [6], it was proven that $w(A) \leq_{\mathrm{p}} w(B)$, which is also a corollary of the above proposition.

Another theorem links monotone inductions with well-founded inductions. One of the symmetric ultimate approximations of a monotone lattice operator $O : L \to L$ is the operator $\mathcal{U}_O : L^2 \to L^2$ which maps (x, y) to $(O(x), O(y))$ [6].

Theorem 2. *For any terminal monotone induction $\langle x_\xi \rangle_{\xi \leq \alpha}$ of O, the sequence $\langle (x_\xi, y_\xi) \rangle_{\xi \leq \alpha+1}$ with $y_\xi = \top$ for every $\xi \leq \alpha$ and $x_{\alpha+1} = y_{\alpha+1} = lfp(O)$, is a terminal well-founded induction of $\mathcal{U}_O$.*

4 Well-founded semantics of ID-logic definitions

We assume familiarity with classical logic. A vocabulary Σ consists of a set of predicate and function symbols. Propositional symbols and constants are 0-ary predicate symbols, respectively function symbols. Terms and FO formulas are defined as usual, and are built inductively from variables, constant and function symbols and logical connectives and quantifiers.

A *definition* is a set of rules of the form

$$\forall \bar{x} \quad (P(\bar{t}) \leftarrow \phi)$$

where ϕ is a FO formula over Σ and $\bar{t}$ is a tuple of terms over Σ such that the free variables of ϕ and the variables of $\bar{t}$ all occur in $\bar{x}$. We call $P(\bar{t})$ the head of the rule, and ϕ the body. The connective $\leftarrow$ is called *definitional implication* and is to be distinguished from material implication $\supset$. A predicate appearing in the head of a rule of a definition Δ is called a *defined predicate* of Δ , any other symbol is called an *open symbol* of Δ. The sets of defined predicates, respectively open symbols of Δ are denoted $Def(\Delta)$, respectively $Open(\Delta) = \Sigma \setminus Def(\Delta)$. For simplicity, we assume that every rule is of the form $\forall \bar{x}\ (P(\bar{x}) \leftarrow \phi)$. Every rule $\forall \bar{x} \quad (P(\bar{t}) \leftarrow \phi)$ can be transformed in an equivalent rule of that form. An FO(ID) (or ID-logic) formula is a boolean combination of FO formulas and definitions. An FO(ID) theory is a set of FO-ID formulas without free variables.

The semantics of the FO(ID) is an integration of standard two-valued FO semantics with the well-founded semantics of definitions. For technical reasons, we need to introduce some concepts from three-valued logic. Consider the set $\mathcal{THREE} = \{\mathbf{f}, \mathbf{u}, \mathbf{t}\}$. The *truth order* $\leq$ on this set is induced by $\mathbf{f} < \mathbf{u} < \mathbf{t}$; the *precision order* $\leq_{\mathrm{p}}$ is induced by $\mathbf{u} <_p \mathbf{f}$ and $\mathbf{u} <_p \mathbf{t}$. Define $\mathbf{f}^{-1} = \mathbf{t}, \mathbf{u}^{-1} = \mathbf{u}, \mathbf{t}^{-1} = \mathbf{f}$.

Given a domain D, a *value* for a n-ary function symbol is a function from D^n to D. A value for an n-ary predicate symbol is a function from D^n to $\mathcal{THREE}$. A Σ-interpretation I consists of a domain D^I, and a value σ^I for each symbol $\sigma \in \Sigma$. A two-valued interpretation is one in which predicates have range $\{\mathbf{f}, \mathbf{t}\}$. For each interpretation F for the function symbols of Σ, both truth and precision order have a pointwize extension to an order on all Σ-interpretations extending F.

A domain atom of I is a tuple of a predicate $P \in \Sigma$ and a tuple $(a_1, \ldots, a_n) \in D^n$; it will be denoted $P(a_1, \ldots, a_n)$, or more compactly, $P(\bar{a})$.

For a given Σ-interpretation I, symbol σ and a value v for σ, we denote by $I[\sigma/v]$ the $\Sigma \cup \{\sigma\}$-interpretation, that assigns to all symbols the same value as I, except that $\sigma^{I[\sigma/v]} = v$. Likewise, for a domain atom $P(\bar{a})$ and a truth value $v \in \mathcal{THREE}$, we define $I[P(\bar{a})/v]$ as the interpretation I' identical to I except that $P(\bar{a})^{I'} = P^{I'}(\bar{a}) = v$. Similarly, for any set U of domain atoms, $I[U/v]$ is identical to I except that all atoms in U have value v.

When all symbols of term t are interpreted in I, we define its value t^I using the standard induction. The truth value φ^I of an FO sentence φ in I is defined by induction on the subformula order:

- $P(t_1, \ldots, t_n)^I := P^I(t_1^I, \ldots, t_n^I)$;
- $(\psi \wedge \phi)^I := Min_{\leq}(\psi^I, \phi^I)$;
- $(\neg\psi)^I := (\psi^I)^{-1}$;
- $(\exists x\ \psi)^I = Max_{\leq}(\{\psi^{I[x/d]} \mid d \in D^I\})$.

We now define the semantics of definitions. Let Δ be a definition over Σ and O a two-valued $Open(\Delta)$-interpretation. Consider the collection $\mathcal{V}_O^\Sigma$ of three-valued Σ-structures extending O. On this set, we define the three-valued immediate consequence operator Ψ_Δ^O, also called the Fitting operator, which maps any $I \in \mathcal{V}_O^\Sigma$ to the O-extension J such that for each defined domain atom $P(\bar{a})$,

$$P(\bar{a})^J = Max_{\leq}(\{\varphi(\bar{a})^J | \forall \mathbf{x}(P(\mathbf{x}) \leftarrow \varphi) \in \Delta\}).$$

The Fitting operator [11] is the extension of the van Emden-Kowalski operator to three-valued structures.

Let L be the lattice of two-valued Σ-structures extending O. As shown in [6], $\mathcal{V}_O^\Sigma$ is isomorphic with L^c and the correspondence is between three-valued interpretations K and tuples of two-valued interpretations (I, J) such that for each domain atom $P(\bar{a})$,

$$\begin{cases} P(\bar{a})^K = \mathbf{t} \text{ and } P(\bar{a})^I = \mathbf{t} = P(\bar{a})^J; \\ P(\bar{a})^K = \mathbf{u} \text{ and } P(\bar{a})^I = \mathbf{f}, P(\bar{a})^J = \mathbf{t}; \\ P(\bar{a})^K = \mathbf{f} \text{ and } P(\bar{a})^I = \mathbf{f} = P(\bar{a})^J. \end{cases}$$

We denote the two components of a three-valued K by K_1 and K_2. In this view, the Fitting operator is an exact L^c-approximation of the van Emden-Kowalski operator, and has a well-founded fixpoint, denoted I_o^{Δ}. This is, in general, a three-valued structure. We extend the truth valuation function φ^I to all FO(ID) formulas by extending the above recursive rules with a new base case for definitions. For a given three-valued structure I and definition Δ, we define $\Delta^I = \mathbf{t}$ if $I = (I|_{Open(\Delta)})^{\Delta}$, and $\Delta^I = \mathbf{f}$ otherwise.

We are now ready to define the semantics of FO(ID). A structure I satisfies a FO(ID) sentence φ (is a model of φ) if I is two-valued and $\varphi^I = \mathbf{t}$. As usual, this is denoted $I \models \varphi$. I satisfies a FO(ID) theory T if I satisfies every $\varphi \in T$. Note that the semantics is two-valued and extends the semantics of classical logic. The restriction to consider only two-valued well-founded models boils down to the requirement that a definition Δ should be *total*, i.e., should define the truth of all defined domain atoms (see [8]).

We now apply the results of the previous section to derive an alternative definition of the well-founded model, which is simpler, more intuitive and more flexible than the one above. We first generalize the well-known concept of an unfounded set [23].

Definition 5. *Given a definition Δ and a three-valued Σ-structure I, an unfounded set of Δ in I is a non-empty set U of defined domain atoms such that each $P(\bar{a}) \in U$ is unknown in I and for each rule $\forall \bar{x}\ (P(\bar{x}) \leftarrow \phi(\bar{x})) \in \Delta$, $\phi(\bar{a})^{I[U/\mathbf{f}]} = \mathbf{f}$.*

When U is an unfounded set in an interpretation I which corresponds to a pair (J, K), then $I[U/\mathbf{f}]$ corresponds to $(J, K[U/\mathbf{f}])$. If, in addition, I is Ψ_{Δ}^{O}-contracting, then it is easy to see that each domain atom $P(\bar{a})$, false in $I[U/\mathbf{f}]$, is false in $\Psi_{\Delta}^{O}(I[U/\mathbf{f}])$, or, equivalently, $\Psi_{\Delta}^{O}(I[U/\mathbf{f}])_2 \leq I[U/\mathbf{f}]_2 = K[U/\mathbf{f}]$. Hence, $I[U/\mathbf{f}]$ is a Ψ_{Δ}^{O}-refinement of I.

Definition 6. *We define a well-founded induction of a definition Δ in an $Open(\Delta)$-interpretation O as a sequence $\langle I^{\xi} \rangle_{\xi \leq \alpha}$ of three-valued Σ-structures extending O such that:*

- *for every defined predicate symbol P, P^{I^0} is the constant function* $\mathbf{u}$*,*
- *for each limit ordinal $\lambda \leq \alpha$, $I^{\lambda} = lub_{\leq_p}(\{I^{\xi} \mid \xi < \lambda\})$, and*
- *for every ordinal ξ, $I^{\xi+1}$ relates to I^{ξ} in one of the following ways.*
  - *$I^{\xi+1} := I^{\xi}[P(\bar{a})/\mathbf{t}]$, for some domain atom $P(\bar{a})$, unknown in I^{ξ}, such that for some rule $\forall \bar{x}\ (P(\bar{x}) \leftarrow \phi(\bar{x})) \in \Delta$, $\phi(\bar{a})^{I^{\xi}} = \mathbf{t}$;*
  - *$I^{\xi+1} := I^{\xi}[U/\mathbf{f}]$, where U is an unfounded set of Δ in I^{ξ}.*

A well-founded induction is terminal *if it cannot be extended anymore.*

We will call an interpretation $I[P(\bar{a})/\mathbf{t}]$ or $I[U/\mathbf{f}]$ satisfying the conditions in the above definition a Δ-refinement of I in O.

In such a sequence, for each $\xi < \alpha$, it either holds that $I^{\xi} <_p I^{\xi+1} = I^{\xi}[P(\bar{a})/\mathbf{t}] \leq_p \Psi_{\Delta}^{O}(I^{\xi})$, or $I^{\xi+1} = I^{\xi}[U/\mathbf{f}]$ with U an unfounded set. It follows that $I^{\xi+1}$ is a Ψ_Δ^O-refinement. Hence, each well-founded induction of Δ in O is a well-founded induction of Ψ_Δ^O. The inverse is clearly not the case (in an induction of Ψ_Δ^O, many atoms can be made true at the same time). Still, a terminal well-founded induction of Δ with limit I^α is a terminal induction of Ψ_Δ^O. Indeed, suppose I^α has a Ψ_Δ^O-refinement J. Then either it must be that I^α has an unfounded set U, or $I^\alpha <_p \Psi_\Delta^O(I^\alpha)$ which implies that for at least one domain atom $P(\bar{a})$, $P(\bar{a})^I = \mathbf{u}$ while $P(\bar{a})^{\Psi_\Delta^O(I^\alpha)} = \mathbf{t}$ or $P(\bar{a})^{\Psi_\Delta^O(I^\alpha)} = \mathbf{f}$. In the latter case, $\{P(\bar{a})\}$ is an unfounded set. In all cases, I^α has a Δ-refinement.

Proposition 9. *A (terminal) well-founded induction of definition Δ in O is a (terminal) well-founded induction of the approximator Ψ_Δ^O.*

Therefore, the results of the previous section now directly yield following theorem, which gives us a characterization of the well-founded model as the limit of *any* well-founded induction.

Theorem 3. *There exist terminal well-founded inductions of Δ in O. Each well-founded induction of Δ in O is strictly increasing in precision. The limit of every terminal well-founded induction of Δ in O is the well-founded model O^Δ.*

5 Conclusion

Approximation theory is an extension of Tarski's least-fixpoint theorem of monotone lattice operators [21] to the case of arbitrary ones. The claim has been made that this theory is the (missing) fixpoint theory of generalized non-monotone forms of induction such as induction over a well-founded order and iterated induction. In this paper, we gave an argument for this, by investigating a natural class of constructive processes and showing that these are confluent, all having the well-founded model as their limit. This result allowed us to derive a new, simpler and more elegant definition of the well-founded semantics of rule sets, that does not rely on the immediate consequence operator. It would also allow to derive new, simpler constructive characterisations of the well-founded semantics of default logic and auto-epistemic logic. As we have argued in the introduction, this definition also provides a better model of what happens in current implementations of the well-founded semantics.

References

1. P. Aczel. An introduction to inductive definitions. In J. Barwise, editor, *Handbook of Mathematical Logic*, pages 739–782. North-Holland Publishing Company, 1977.
2. Marc Denecker. Extending classical logic with inductive definitions. In *First International Conference on Computational Logic (CL'2000)*, volume 1861 of *Lecture Notes in Artificial Intelligence*, pages 703–717. Springer, 2000.
3. Marc Denecker, Maurice Bruynooghe, and Victor Marek. Logic programming revisited: Logic programs as inductive definitions. *ACM Transactions on Computational Logic*, 2(4):623–654, October 2001.

4. Marc Denecker, Victor Marek, and Mirosław Truszczyński. Approximating operators, stable operators, well-founded fixpoints and applications in nonmonotonic reasoning. In J. Minker, editor, *Logic-based Artificial Intelligence*, chapter 6, pages 127–144. Kluwer Academic Publishers, 2000.
5. Marc Denecker, Victor Marek, and Mirosław Truszczyński. Uniform semantic treatment of default and autoepistemic logics. *Artificial Intelligence*, 143(1):79–122, 2003.
6. Marc Denecker, Victor Marek, and Mirosław Truszczyński. Ultimate approximation and its application in nonmonotonic knowledge representation systems. *Information and Computation*, 192(1):84–121, 2004.
7. Marc Denecker and Eugenia Ternovska. A logic of non-monotone inductive definitions and its modularity properties. In Vladimir Lifschitz and Ilkka Niemelä, editors, *Seventh International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'7)*, 2004.
8. Marc Denecker and Eugenia Ternovska. A logic of non-monotone inductive definitions. *Transactions On Computational Logic (TOCL)*, 2007.
9. H.-D. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer-Verlag, 1999.
10. S. Feferman. Formal theories for transfinite iterations of generalised inductive definitions and some subsystems of analysis. In A. Kino, J. Myhill, and R.E. Vesley, editors, *Intuitionism and Proof theory*, pages 303–326. North Holland, 1970.
11. M. Fitting. A Kripke-Kleene Semantics for Logic Programs. *Journal of Logic Programming*, 2(4):295–312, 1985.
12. M. Fitting. The family of stable models. *Journal of Logic Programming*, 17(2,3&4):197–225, 1993.
13. Zbigniew Lonc and Mirosław Truszczyński. On the problem of computing the well-founded semantics. *Theory and practice of Logic Programming*, 1(5):591–609, 2001.
14. Maarten Mariën, Rudradeb Mitra, Marc Denecker, and Maurice Bruynooghe. Satisfiability checking for PC(ID). In Geoff Sutcliffe and Andrei Voronkov, editors, *LPAR'05*, volume 3835 of *Lecture Notes in Computer Science*, pages 565–579. Springer, 2005.
15. David Mitchell and Eugenia Ternovska. A framework for representing and solving np search problems. In *AAAI'05*, pages 430–435. AAAI Press/MIT Press, 2005.
16. Y. N. Moschovakis. *Elementary Induction on Abstract Structures*. North-Holland Publishing Company, Amsterdam- New York, 1974.
17. Y. N. Moschovakis. On non-monotone inductive definability. *Fundamenta Mathematica*, 82(39-83), 1974.
18. Ilkka Niemelä, Patrik Simons, and Tommi Syrjänen. Smodels: a system for answer set programming. In *Proceedings of the 8th International Workshop on Non-Monotonic Reasoning*, Breckenridge, Colorado, USA, April 2000. cs.AI/0003033.
19. E. Post. Formal reduction of the general combinatorial decision problem. *American Journal of Mathematics*, 65:197–215, 1943.
20. C. Spector. Inductively defined sets of natural numbers. In *Infinitistic Methods (Proc. 1959 Symposium on Foundation of Mathematis in Warsaw)*, pages 97–102. Pergamon Press, Oxford, 1961.
21. A. Tarski. Lattice-theoretic fixpoint theorem and its applications. *Pacific journal of Mathematics*, 5:285–309, 1955.
22. Bas C. van Fraassen. Singular Terms, truth-Value Gaps, and Free Logic. *The journal of Philosophy*, 63(17):481–495, 1966.
23. Allen Van Gelder, Kenneth A. Ross, and John S. Schlipf. The well-founded semantics for general logic programs. *Journal of the ACM*, 38(3):620–650, 1991.