On the Relation Between Approximation Fixpoint Theory and Justification Theory

Simon Marynissen[1,2] , **Bart Bogaerts**[2] and **Marc Denecker**[1]
[1]KU Leuven
[2]Vrije Universiteit Brussel
simon.marynissen@kuleuven.be, bart.bogaerts@vub.be, marc.denecker@kuleuven.be

Abstract

Approximation Fixpoint Theory (AFT) and Justification Theory (JT) are two frameworks to unify logical formalisms. AFT studies semantics in terms of fixpoints of lattice operators, and JT in terms of so-called justifications, which are explanations of why certain facts do or do not hold in a model. While the approaches differ, the frameworks were designed with similar goals in mind, namely to study the different semantics that arise in (mainly) non-monotonic logics. The first contribution of our current paper is to provide a formal link between the two frameworks. To be precise, we show that every justification frame induces an approximator and that this mapping from JT to AFT preserves all major semantics. The second contribution exploits this correspondence to extend JT with a novel class of semantics, namely *ultimate semantics*: we formally show that ultimate semantics can be obtained in JT by a syntactic transformation on the justification frame, essentially performing some sort of resolution on the rules.

1 Introduction

In this framework, we are concerned with two theories developed with similar intentions, namely to unify semantics of (mostly non-monotonic) logics, namely Approximation Fixpoint Theory (AFT) and Justification Theory (JT).

Approximation Fixpoint Theory

In the 1980s and 90s, the area of non-monotonic reasoning (NMR) saw fierce debates about formal semantics. In several subareas, researchers sought to formalize common-sense intuitions about knowledge of introspective agents. In these areas, appeals to similar intuitions were made, resulting in the development of similar mathematical concepts. Despite the obvious similarity, the precise relation between these concepts remained elusive. AFT was founded in the early 2000s by Denecker, Marek and Truszczyński [2000] as a way of unifying semantics that emerged in these different subareas. The main contribution of AFT was to demonstrate that, by moving to an algebraic setting, the common principles behind these concepts can be isolated and studied in a general way. This breakthrough allowed results that were achieved in the context of one of these languages to be easily transferred to another. In the early stages, AFT was applied to default logic, auto-epistemic logic, and logic programming [Denecker *et al.*, 2000; Denecker *et al.*, 2003]. In recent years also applications in various other domains have emerged [Strass, 2013; Bi *et al.*, 2014; Charalambidis *et al.*, 2018; Bogaerts and Cruz-Filipe, 2018].

The foundations of AFT lie in Tarski's fixpoint theory of monotone operators on a complete lattice [Tarski, 1955]. AFT demonstrates that by moving from the original lattice L to the bilattice L^2, Tarski's theory can be generalized into a fixpoint theory for arbitrary (i.e., also non-monotone) operators. Crucially, all that is required to apply AFT to a formalism and obtain several semantics is to define an appropriate approximating operator $L^2 \to L^2$ on the bilattice; the algebraic theory of AFT then takes care of the rest. For instance, to characterize the major logic programming semantics using AFT, it suffices to define Fitting's four-valued immediate consequence operator [Fitting, 2002]. The (partial) stable fixpoints of that operator (as defined by AFT) are exactly the partial stable models of the original program; the well-founded fixpoint of the operator is the well-founded model of the program, etc.

Justification Theory

Building on an old semantical framework for (abductive) logic programming [Denecker and De Schreye, 1993], Denecker *et al.* [2015] defined an abstract theory of *justifications* suitable for describing the semantics of a range of logics in knowledge representation, computational and mathematical logic, including logic programs, argumentation frameworks and nested least and greatest fixpoint definitions. Justifications provide a refined way of describing the semantics of a logic: they not only define whether an interpretation is a model (under a suitable semantics) of a theory, but also *why*.

Justifications — albeit not always in the exact formal form as described by Denecker *et al.* [2015] — have appeared in different ways in different areas. The stable semantics for logic programs was defined in terms of justifications [Fages, 1990; Schulz and Toni, 2013]. Moreover, an algebra for combining justifications (for logic programs) was defined by Cabalar *et al.* [2014]; and justifications are underlying provenance systems in databases [Damásio *et al.*, 2013].

Next to these theoretic benefits, justifications are also used in implementations of answer set solvers (they form the basis of the so-called source-pointer approach in the unfounded set algorithm [Gebser *et al.*, 2009], and turned out to be key in analyzing conflicts in the context of lazy grounding [Bogaerts and Weinzierl, 2018]), as well as to improve parity game solvers [Lapauw *et al.*, 2020].

Correspondence

The two described frameworks were designed with similar intentions in mind, namely to unify different (mainly non-monotonic) logics. One major difference between them is that JT is defined logically while AFT is defined purely algebraically. This makes justification frameworks less abstract and easier to grasp, but also in a certain sense less general. On the other hand, Denecker *et al.* [2015] defined a notion of nesting, which seems promising to integrate the semantics of nested least and nested greatest fixpoint definitions.

Despite the differences, certain correspondences between the theories show up: several definitions in justification frameworks seem to have an algebraical counterpart in AFT. This is evident from the fact that many results on justifications are formulated in terms of fixpoints of a so-called derivation operator that happens, for the case of logic programming, to coincide with (Fitting's three-valued version of) the immediate consequence operator for logic programs. Of course, now the question naturally arises whether this correspondence can be made formal, i.e., whether it can formally be shown that semantics induced by JT will always coincide with their equally-named counterpart in AFT. If the answer is positive, this will allow us to translate results between the two theories. Formalizing this correspondence is the key contribution of the current paper.

Contributions

Our contributions can be summarized as follows:

- In Section 3, we provide some novel results for JT. While the main purpose of these results is to support the theorems of Section 4, they also directly advance the state of JT. In this section, we show among others how different semantics induced by JT relate, and we resolve a discrepancy that exists between different definitions of so-called *stable and supported branch evaluations* in prior work. We formally prove that the different circulating definitions of these branch evaluations indeed induce the same semantics.
- In Section 4, we turn our attention to the key contribution of the paper, namely embedding JT in AFT. To do this, we proceed as follows. First, we show that under minor restrictions, each justification frame (intuitively, this is a set of rules that describe when a positive or negative fact is true), can be transformed into an approximator. Next, we show that for each of the most common *branch evaluations* (these are mathematical structures that are used to associate semantics to a justification frame), the induced semantics by JT is the same as the equally-named semantics on the AFT side. Establishing this result is of particular importance for the future development of JT, since this result immediately makes a large body of theoretical results developed in the context of AFT readily available for JT, as well as all its future application domains, including results on stratification [Vennekens *et al.*, 2006; Bogaerts and Cruz-Filipe, 2021], predicate introduction [Vennekens *et al.*, 2007], and knowledge compilation [Bogaerts and Van den Broeck, 2015]. On the other hand, from the context of AFT, the embedding of JT can serve as inspiration for developing more general algebraic explanation mechanisms.
- To illustrate how this connection can be exploited for further exploiting the theory of justifications, we turn our attention to *ultimate semantics*. In the context of AFT, Denecker and his coauthors have realized that a single operator can have multiple approximators and that the choice of approximator influences the induced semantics. They also showed that — when staying in the realm of consistent AFT — every operator induces a most precise approximator, and called this *the ultimate approximator* [Denecker *et al.*, 2004]. In Section 5, we transfer this idea to JT. We show there that by means of a simple transformation[1] on the justification frame, we can obtain ultimate semantics. Importantly, since this transformation is defined independent of the branch evaluation at hand, ultimate semantics are not just induced for the semantics that have a counterpart in AFT, but for all conceivable current and future branch evaluations as well.

2 Preliminaries: Justification Theory

In this section we use the formalization of JT as done by Marynissen *et al.* [2020]. Truth values are denoted $\mathbf{t}$ (true), $\mathbf{f}$ (false) and $\mathbf{u}$ (unknown); we write $\mathcal{L}$ for $\{\mathbf{t}, \mathbf{f}, \mathbf{u}\}$. We make use of two orders on $\mathcal{L}$, the *truth order* $\mathbf{f} \leq_t \mathbf{u} \leq_t \mathbf{t}$ and the *precision order* $\mathbf{u} \leq_p \mathbf{f}, \mathbf{t}$. JT starts with a set $\mathcal{F}$, referred to as a *fact space*, such that $\mathcal{L} \subseteq \mathcal{F}$; the elements of $\mathcal{F}$ are called *facts*. We assume that $\mathcal{F}$ is equipped with an involution $\sim: \mathcal{F} \to \mathcal{F}$ (i.e., a bijection that is its own inverse) such that $\sim\mathbf{t} = \mathbf{f}$, $\sim\mathbf{u} = \mathbf{u}$, and $\sim x \neq x$ for all $x \neq \mathbf{u}$. Moreover, we assume that $\mathcal{F} \setminus \mathcal{L}$ is partitioned into two disjoint sets $\mathcal{F}_+$ and $\mathcal{F}_-$ such that $x \in \mathcal{F}_+$ if and only if $\sim x \in \mathcal{F}_-$ for all $x \in \mathcal{F} \setminus \mathcal{L}$. Elements of $\mathcal{F}_+$ are called *positive* and elements of $\mathcal{F}_-$ are called *negative* facts. An example of a fact space is the set of literals over a propositional vocabulary Σ extended with $\mathcal{L}$ where $\sim$ maps a literal to its negation. For any set A we define $\sim A$ to be the set of elements of the form $\sim a$ for $a \in A$. We distinguish two types of facts: *defined* and *open* facts. The former are accompanied by a set of rules that determine their truth value. The truth value of the latter is not governed by the rule system but comes from an external source or is fixed (as is the case for logical facts).

Definition 1. A *justification frame* $\mathcal{JF}$ is a tuple $\langle \mathcal{F}, \mathcal{F}_d, R \rangle$ such that

[1] Essentially, this transformation performs some sort of case splitting.

- $\mathcal{F}_d$ is a subset of $\mathcal{F}$ closed under $\sim$, i.e., $\sim\mathcal{F}_d = \mathcal{F}_d$; facts in $\mathcal{F}_d$ are called *defined*;
- no logical fact is defined: $\mathcal{L} \cap \mathcal{F}_d = \emptyset$;
- $R \subseteq \mathcal{F}_d \times 2^{\mathcal{F}\setminus\emptyset}$;
- for each $x \in \mathcal{F}_d$ there is at least one element $(x, A) \in R$.

The set $\mathcal{F}_o$ of *open* facts is equal to $\mathcal{F} \setminus \mathcal{F}_d$. An element $(x, A) \in R$ is called a *rule* with *head* x and *body* (or *case*) A. The set of cases of x in $\mathcal{JF}$ is denoted $\mathcal{JF}(x)$. Rules $(x, A) \in R$ are denoted as $x \leftarrow A$ and if $A = \{y_1, \ldots, y_n\}$, we often write $x \leftarrow y_1, \ldots, y_n$.

Logic programming rules can easily be transfered to rules in a justification frame. However, in logic programming, only rules for positive facts are given; never for negative facts. Hence, in order to apply JT to logic programming, a mechanism for deriving rules for negative literals is needed. For this, a technique called *complementation* was invented [Denecker *et al.*, 2015]; it is a generic mechanism that allows turning a set of rules for x into a set of rules for $\sim x$. To define complementation, we first define *selection functions* for x. A selection function for x is a mapping $s\colon \mathcal{JF}(x) \to \mathcal{F}$ such that $s(A) \in A$ for all rules of the form $x \leftarrow A$. Intuitively, a selection function chooses an element from the body of each rule of x. For a selection function s, the set $\{s(A) \mid A \in \mathcal{JF}(x)\}$ is denoted by $\mathrm{Im}(s)$.

Definition 2. For a set of rules R, we define R^* to be the set of rules of the form $\sim x \leftarrow \sim Im(s)$ for $x \in \mathcal{F}_d$ that has rules in R and s a selection function for x. The *complementation* of $\mathcal{JF}$ is defined as $\langle \mathcal{F}, \mathcal{F}_d, R \cup R^* \rangle$. A justification frame $\mathcal{JF}$ is *complementary* if it is fixed under complementation, i.e., $R \cup R^* = R$.

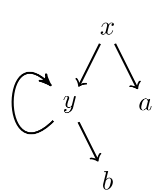
Example 1. *If* $R = \{x \leftarrow a, b;\ x \leftarrow c;\}$*, then* $R^* = \{\sim x \leftarrow \sim a, \sim c;\ \sim x \leftarrow \sim b, \sim c\}$.

Definition 3. A *directed graph* is a pair (N, E) where N is a set of nodes and $E \subseteq N \times N$ is the set of edges. An *internal* node is a node with outgoing edges. A *leaf* is a non-internal node.

Definition 4. Let $\mathcal{JF} = \langle \mathcal{F}, \mathcal{F}_d, R \rangle$ be a justification frame. A *justification* J in $\mathcal{JF}$ is a directed graph (N, E) such that for every internal node $n \in N$ it holds that $n \leftarrow \{m \mid (n, m) \in E\} \in R$;

A justification is *locally complete* if it has no leaves in $\mathcal{F}_d$. We write $\mathfrak{J}(x)$ to denote the set of locally complete justifications that have an internal node x.

Example 2. *Take* $\mathcal{F}_d = \{x, \sim x, y, \sim y\}$, $\mathcal{F}_o = \{a, \sim a, b, \sim b\} \cup \mathcal{L}$*, and* R *the complementation of* $\{x \leftarrow y, a; y \leftarrow y, b\}$*, then*



is a locally complete justification in $\langle \mathcal{F}, \mathcal{F}_d, R \rangle$ *because* a *and* b *are open facts.*

Definition 5. Let $\mathcal{JF}$ be a justification frame. A $\mathcal{JF}$*-branch* is either an infinite sequence in $\mathcal{F}_d$ or a finite non-empty sequence in $\mathcal{F}_d$ followed by an element in $\mathcal{F}_o$. For a justification J in $\mathcal{JF}$, a J*-branch* starting from $x \in \mathcal{F}_d$ is a path in J starting from x that is either infinite or ends in a leaf of J. We write $B_J(x)$ to denote the set of J-branches starting from x.

Not all J-branches are $\mathcal{JF}$-branches since they can end in a defined fact. However, if J is locally complete, any J-branch is also a $\mathcal{JF}$-branch. We denote a branch $\mathbf{b}$ as $\mathbf{b}\colon x_0 \to x_1 \to \cdots$ and define $\sim\mathbf{b}$ as $\sim x_0 \to \sim x_1 \to \cdots$. A *tail* of a branch $\mathbf{b}$ is a branch $x_i \to x_{i+1} \to \cdots$ for some $i \geq 0$.

Definition 6. A *branch evaluation* $\mathcal{B}$ is a mapping that maps any $\mathcal{JF}$-branch to an element in $\mathcal{F}$ for all justification frames $\mathcal{JF}$. A branch evaluation $\mathcal{B}$ is *consistent* if $\mathcal{B}(\sim\mathbf{b}) = \sim\mathcal{B}(\mathbf{b})$ for any branch $\mathbf{b}$. A justification frame $\mathcal{JF}$ together with a branch evaluation $\mathcal{B}$ form a *justification system* $\mathcal{JS}$, which is presented as a quadruple $\langle \mathcal{F}, \mathcal{F}_d, R, \mathcal{B} \rangle$.

The main branch evaluations we are interested in are given below:

Definition 7. The *supported* branch evaluation $\mathcal{B}_{\mathrm{sp}}$ maps $x_0 \to x_1 \to \cdots$ to x_1. The *Kripke-Kleene* branch evaluation $\mathcal{B}_{\mathrm{KK}}$ maps finite branches to their last element and infinite branches to $\mathbf{u}$. The *well-founded* branch evaluation $\mathcal{B}_{\mathrm{wf}}$ maps finite branches to their last element. It maps infinite branches to $\mathbf{t}$ if they have a negative tail, to $\mathbf{f}$ if they have a positive tail and to $\mathbf{u}$ otherwise. The *stable* branch evaluation $\mathcal{B}_{\mathrm{st}}$ maps a branch $x_0 \to x_1 \to \cdots$ to the first element that has a different sign than x_0 if it exists; otherwise $\mathbf{b}$ is mapped to $\mathcal{B}_{\mathrm{wf}}(\mathbf{b})$.

Definition 8. A *(three-valued) interpretation* of $\mathcal{F}$ is a function $\mathcal{I}\colon \mathcal{F} \to \mathcal{L}$ such that $\mathcal{I}(\sim x) = \sim\mathcal{I}(x)$ for all $x \in \mathcal{F}$ and $\mathcal{I}(\ell) = \ell$ for all $\ell \in \mathcal{L}$.

We will assume that the interpretation of open facts is fixed; hence any two interpretations coincide on open facts.

Definition 9. Let $\mathcal{JS} = \langle \mathcal{F}, \mathcal{F}_d, R, \mathcal{B} \rangle$ be a justification system, $\mathcal{I}$ an interpretation of $\mathcal{F}$, and J a locally complete justification in $\mathcal{JS}$. Let $x \in \mathcal{F}_d$ be a node in J. The *value* of $x \in \mathcal{F}_d$ by J under I is defined as $\mathrm{val}_{\mathcal{B}}(J, x, \mathcal{I}) = \min_{\mathbf{b} \in B_J(x)} \mathcal{I}(\mathcal{B}(\mathbf{b}))$, where $\min$ is with respect to $\leq_t$.

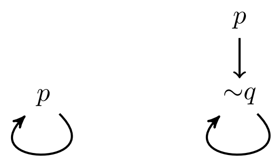
The *supported value* of $x \in \mathcal{F}$ in $\mathcal{JS}$ under $\mathcal{I}$ is defined as $\mathrm{SV}_{\mathcal{JS}}(x, \mathcal{I}) = \max_{J \in \mathfrak{J}(x)} \mathrm{val}_{\mathcal{B}}(J, x, \mathcal{I})$ for $x \in \mathcal{F}_d$ and $\mathrm{SV}(x, \mathcal{I}) = \mathcal{I}(x)$ for $x \in \mathcal{F}_o$. For any $\mathcal{F}$-interpretation $\mathcal{I}$, $\mathcal{S}_{\mathcal{JS}}(\mathcal{I})$ is the mapping $\mathcal{F} \to \mathcal{L}\colon x \mapsto \mathrm{SV}_{\mathcal{JS}}(x, \mathcal{I})$. The function $\mathcal{S}_{\mathcal{JS}}$ is called the *support operator*. If $\mathcal{JS}$ consists of $\mathcal{JF}$ and $\mathcal{B}$, then we write $\mathcal{S}^{\mathcal{B}}_{\mathcal{JF}}$ for $\mathcal{S}_{\mathcal{JS}}$. If $\mathcal{JF}$ is clear from context, we write $\mathrm{SV}_{\mathcal{B}}$ for $\mathrm{SV}_{\mathcal{JS}}$.

Models under justification semantics are determined by the supported value.

Definition 10. Let $\mathcal{JF}$ be justification frame and $\mathcal{B}$ a branch evaluation. An $\mathcal{F}$-interpretation $\mathcal{I}$ is a $\mathcal{B}$*-model of* $\mathcal{JF}$ if for all $x \in \mathcal{F}$, $\mathrm{SV}^{\mathcal{B}}_{\mathcal{JF}}(x, \mathcal{I}) = \mathcal{I}(x)$, i.e., $\mathcal{I}$ is a fixpoint of $\mathcal{S}^{\mathcal{B}}_{\mathcal{JF}}$.

A $\mathcal{B}_{\mathrm{sp}}$, $\mathcal{B}_{\mathrm{KK}}$, $\mathcal{B}_{\mathrm{st}}$, or $\mathcal{B}_{\mathrm{wf}}$-model is called a supported, Kripke-Kleene, stable, or well-founded model.

Example 3. *Let* $\mathcal{F} = \{p, \sim p, q, \sim q\} \cup \mathcal{L}$ *and take* R *to be the complementation of* $\{p \leftarrow p; p \leftarrow \sim q; q \leftarrow q\}$*, i.e. adding* *the rules $\sim p \leftarrow \sim p, q$ and $\sim q \leftarrow \sim q$. There are exactly two locally complete justifications with p as an internal node:*



Under $\mathcal{B}_{\mathrm{wf}}$, the left justification has a value $\mathbf{f}$ *for p, while the right has a value* $\mathbf{t}$ *for p. Since these justifications are the only locally complete ones containing p as an internal node we have that $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(p, \mathcal{I}) = \mathbf{t}$ for all interpretations $\mathcal{I}$ of $\mathcal{F}$. This shows us that the unique $\mathcal{B}_{\mathrm{wf}}$-model is the interpretation mapping p to $\mathbf{t}$ and q to $\mathbf{f}$.*

3 Tying Up Loose Ends

In this section, we prove some results about JT that will be needed for developing our theory later on. These results resolve several issues that were left open in prior work, but turn out to be crucial for studying the relationship with AFT.

3.1 Pasting Justifications

Our first result is essentially a pasting result. What it states is that we can, for all branch evaluations of interest to the current paper, build a single justification that explains the value of all facts. In other words, it provides a means of gluing justifications for different facts together. The first theorem has already been proven in a different context by Marynissen *et al.* [2018], but since our results heavily depend on it, we provide a proof of it in the appendix.

Theorem 1. *Take $\mathcal{B} \in \{\mathcal{B}_{\mathrm{sp}}, \mathcal{B}_{\mathrm{KK}}, \mathcal{B}_{\mathrm{st}}, \mathcal{B}_{\mathrm{wf}}\}$. For every interpretation $\mathcal{I}$, there is a locally complete justification J such that $\mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I})$ for all $x \in \mathcal{F}_d$.*

Similarly, this holds for $\mathcal{B}'_{\mathrm{sp}}$ and $\mathcal{B}'_{\mathrm{st}}$, but only for models.

Theorem 2. *Take $\mathcal{B} \in \{\mathcal{B}'_{\mathrm{sp}}, \mathcal{B}'_{\mathrm{st}}\}$. For every $\mathcal{B}$-model $\mathcal{I}$, there is a locally complete justification J such that $\mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I})$ for all $x \in \mathcal{F}_d$.*

3.2 Equivalence of Branch Evaluations

Our second result concerns different versions of the stable and supported branch evaluations that circulate in prior work. Marynissen *et al.* [2018; 2020] use the stable and supported branch evaluations as we defined them in Definition 7, while Denecker *et al.* [2015] use the following alternative.

Definition 11. The branch evaluation $\mathcal{B}'_{\mathrm{sp}}$ is equal to $\mathcal{B}_{\mathrm{sp}}$ on infinite branches and maps finite branches to their last element. The branch evaluation $\mathcal{B}'_{\mathrm{st}}$ is equal to $\mathcal{B}_{\mathrm{st}}$ except $\mathcal{B}'_{\mathrm{st}}$ maps any finite branch to its last element.

Of course this begs the question in which sense these branch evaluations are related. What we show next is that for the purpose of defining models, they are interchangeable.

Definition 12. Two branch evaluations $\mathcal{B}_1$ and $\mathcal{B}_2$ are *equivalent* if for all justification frames $\mathcal{JF}$, the $\mathcal{B}_1$-models and the $\mathcal{B}_2$-models of $\mathcal{JF}$ coincide.

Our proofs that $\mathcal{B}_{\mathrm{sp}}$ and $\mathcal{B}'_{\mathrm{sp}}$, and $\mathcal{B}_{\mathrm{st}}$ and $\mathcal{B}'_{\mathrm{st}}$ are equivalent, will make use of the following lemma, which intuitively states that to show that an interpretation is a $\mathcal{B}$-model, it suffices to show that the supported value of each fact is *at least* its value in the interpretation.

Lemma 1. *Take $\mathcal{JS} = \langle \mathcal{F}, \mathcal{F}_d, R, \mathcal{B} \rangle$ with $\mathcal{B}$ consistent. Every interpretation $\mathcal{I}$ such that $\mathrm{SV}_{\mathcal{JS}}(x, \mathcal{I}) \geq_t \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$, is a $\mathcal{B}$-model of $\mathcal{JF}$.*

Proposition 1. *The two supported branch evaluations $\mathcal{B}_{\mathrm{sp}}$ and $\mathcal{B}'_{\mathrm{sp}}$ are equivalent.*

Proposition 2. *The two stable branch evaluations $\mathcal{B}_{\mathrm{st}}$ and $\mathcal{B}'_{\mathrm{st}}$ are equivalent.*

Sketch of the proofs of Propositions 1 and 2. The difference between the two branch evaluations at hand is that in the one ($\mathcal{B}'$), finite branches are evaluated with respect to their final element, and the other ($\mathcal{B}$) with respect to some other element y in the branch (second or first sign switch). Take a $\mathcal{B}$-model $\mathcal{I}$ and take a justification J as from Theorems 1 and 2. With some care, we can prove that the final element of a J-branch has a larger value than first element in $\mathcal{I}$ under $\leq_t$. Therefore, $\mathcal{I}$ satisfies the conditions of Lemma 1, proving that $\mathcal{I}$ is a $\mathcal{B}'$-model. The other direction is proven similarly. □

3.3 Links between Different Justification Models

Our third set of results is concerned with the relation between different semantics induced by JT. In the context of logic programming, it is well-known that there is a unique well-founded model, what the relation between well-founded and stable model is, etcetera. Several such results will follow immediately by establishing the correspondence with AFT, but some of them will be needed in our proof. They are given explicitly, and sometimes in higher generality, in the current section.

First of all, in logic programming, it is well-known that the well-founded and Kripke-Kleene semantics induce a single model. In JT, we find an analogous result for a broad class of branch evaluations. A branch evaluation $\mathcal{B}$ is called *parametric* if $\mathcal{B}(\mathbf{b}) \in \mathcal{F}_o$ for all $\mathcal{JF}$-branches and all justification frames $\mathcal{JF}$. Denecker *et al.* [2015] provided the following result.

Proposition 3. *If $\mathcal{JF}$ is a justification frame and $\mathcal{B}$ a parametric branch evaluation, then $\mathcal{JF}$ has a single $\mathcal{B}$-model.*

In this proposition, we of course make use of our earlier assumption that the value of the open facts is fixed. In general, every interpretation of the open facts induces a single model.

Corollary 1. *Every justification frame has a unique $\mathcal{B}_{\mathrm{KK}}$-model and a unique $\mathcal{B}_{\mathrm{wf}}$-model.*

Proposition 4. *The unique $\mathcal{B}_{\mathrm{KK}}$-model is a $\mathcal{B}_{\mathrm{sp}}$-model.*

Proposition 5. *The well-founded model of $\mathcal{JF}$ is a stable model of $\mathcal{JF}$.*

Proposition 6. *Every stable justification model is a supported justification model.*

Proof sketches of Proposition 4, 5 and 6. The idea of the proof is that the justification according to Theorem 1 also works for the other branch evaluation. □

Lemma 2. *Let $\mathcal{I}$ be a stable justification model. If $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) = \mathbf{f}$, then $\mathcal{I}(x) = \mathbf{f}$. If $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) = \mathbf{t}$, then $\mathcal{I}(x) = \mathbf{t}$.*

Proof sketch. Take $x \in \mathcal{F}_d$ with $\mathcal{I}(x) = \mathbf{f}$. By using that $\mathcal{I}$ is a $\mathcal{B}_{\mathrm{st}}$-model, we can prove that every justification J has a branch $\mathbf{b} \in B_J(x)$ such that $\mathcal{I}(\mathcal{B}_{\mathrm{wf}}(\mathbf{b})) \leq_t \mathbf{u}$. This concludes that $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) \neq \mathbf{t}$. □

Proposition 7. *The well-founded model is the $\leq_p$-least stable model.*

Proof sketch. Lemma 2 implies that the well-founded model is less precise than any stable model. Then Proposition 5 finishes the proof. □

4 Embedding JT in AFT

We now turn our attention to the main topic of this paper, namely formally prove the correspondence between JT and AFT. We start with a brief recall of the basic definitions that constitute AFT, next show how to obtain an approximator out of a justification frame, and finally prove that indeed, all major semantics are preserved under this correspondence.

4.1 Preliminaries: AFT

Given a complete lattice $\langle L, \leq \rangle$, AFT [Denecker *et al.*, 2000] uses the *bilattice* $L^2 = L \times L$. We define projection functions as usual: $(x, y)_1 = x$ and $(x, y)_2 = y$. Pairs $(x, y) \in L^2$ are used to approximate elements in the interval $[x, y] = \{z \mid x \leq z \leq y\}$. We call $(x, y) \in L^2$ *consistent* if $x \leq y$, i.e., if $[x, y]$ is not empty. The set of consistent elements is denoted L^c. A pair (x, x) is called *exact* since it approximates only the element x. The *precision order* $\leq_p$ on L^2 is defined as $(x, y) \leq_p (u, v)$ if $x \leq u$ and $y \geq v$. If (u, v) is consistent, this means that $[u, v] \subseteq [x, y]$. If $\langle L, \leq \rangle$ is a complete lattice, then so is $\langle L^2, \leq_p \rangle$. AFT studies fixpoints of operators $O\colon L \to L$ through operators approximating O. An operator $A\colon L^2 \to L^2$ is an *approximator* of O if it is $\leq_p$-monotone and has the property that $A(x, x) = (O(x), O(x))$ for all $x \in L$. Approximators are internal in L^c (i.e., map L^c into L^c). We often restrict our attention to *symmetric* approximators: approximators A such that, for all x and y, $A(x, y)_1 = A(y, x)_2$. Denecker *et al.* [2004] showed that the consistent fixpoints of interest of a symmetric approximator are uniquely determined by an approximator's restriction to L^c and hence, that it usually suffices to define approximators on L^c. Such a restriction is called a *consistent approximator*. As mentioned before, AFT studies fixpoints of O using fixpoints of A. The main type of fixpoints that concern us are given here.

- A *partial supported fixpoint* of A is a fixpoint of A.
- The *Kripke-Kleene fixpoint* of A is the $\leq_p$-least fixpoint of A; it approximates all fixpoints of A.
- A *partial stable fixpoint* of A is a pair (x, y) such that $x = \mathrm{lfp}(A(\cdot, y)_1)$ and $y = \mathrm{lfp}(A(x, \cdot)_2)$, where $A(\cdot, y)_1$ denotes the function $L \to L\colon z \mapsto A(z, y)_1$ and analogously $A(x, \cdot)_2$ stands for $L \to L\colon z \mapsto A(x, z)_2$.
- The *well-founded fixpoint* of A is the $\leq_p$-least partial stable fixpoint of A.

4.2 The Approximator

Let $\mathcal{JF} = \langle \mathcal{F}, \mathcal{F}_d, R \rangle$ be a justification frame, fixed throughout this section. Our first goal is to define, from a given justification frame, an approximator on a suitable lattice. Following the correspondence with how this is done in logic programming, we will take as lattice the set of *exact* interpretations (interpretations that map no facts to $\mathbf{u}$ except for $\mathbf{u}$ itself). It is easy to see that such interpretations correspond directly to subsets of $\mathcal{F}_+$. In other words, we will use the lattice $\langle L = 2^{\mathcal{F}_+}, \subseteq \rangle$. Now, the set L^c is isomorphic to the set of three-valued interpretations of $\mathcal{F}$; under this isomorphism, a consistent pair $(I, J) \in L^c$ corresponds to the three-valued interpretation $\mathcal{I}$ such that for positive facts $x \in \mathcal{F}_+, \mathcal{I}(x) = \mathbf{t}$ if $x \in I$, $\mathcal{I}(x) = \mathbf{f}$ if $x \notin J$, and $\mathcal{I}(x) = \mathbf{u}$ otherwise.

Definition 13. The *operator* $O_{\mathcal{JF}}\colon L \to L$ of $\mathcal{JF}$ maps a subset I of $\mathcal{F}_+$ to

$$O_{\mathcal{JF}}(I) = \{x \in \mathcal{F}_+ \mid \exists x \leftarrow A \in R\colon \forall a \in A\colon (I, I)(a) = \mathbf{t}\}.$$

The *approximator* $A_{\mathcal{JF}}\colon L^c \to L^c$ of $\mathcal{JF}$ is defined as follows

$$A_{\mathcal{JF}}(\mathcal{I})_1 = \{x \in \mathcal{F}_+ \mid \exists x \leftarrow A \in R\colon \forall a \in A\colon \mathcal{I}(a) = \mathbf{t}\}$$
$$A_{\mathcal{JF}}(\mathcal{I})_2 = \{x \in \mathcal{F}_+ \mid \exists x \leftarrow A \in R\colon \forall a \in A\colon \mathcal{I}(a) \geq_t \mathbf{u}\}$$

Proposition 8. *If no rule body in $\mathcal{JF}$ contains $\mathbf{u}$, then $A_{\mathcal{JF}}$ is a consistent approximator of $O_{\mathcal{JF}}$.*

So far, we are not aware of practical examples with bodies containing $\mathbf{u}$. From now on, we assume that every justification frame does not have $\mathbf{u}$ in a rule body. It turns out that in case our justification frame behaves well with respect to negation (if it is complementary), the approximator is the same operator as induced by the branch evaluation $\mathcal{B}_{\mathrm{sp}}$.

Lemma 3. *For a complementary justification frame $\mathcal{JF}$, the function $A_{\mathcal{JF}}$ and the support operator $\mathcal{S}^{\mathcal{B}_{\mathrm{sp}}}_{\mathcal{JF}}$ are equal.*

4.3 Semantic Correspondence

The central result of this section is the following theorem, which essentially states that for all major semantics, the branch evaluation in JT corresponds to the definitions of AFT.

Theorem 3. *Take a complementary justification frame $\mathcal{JF}$.*

- *The partial supported fixpoints of $A_{\mathcal{JF}}$ are exactly the supported models of $\mathcal{JF}$.*
- *The Kripke-Kleene fixpoint of $A_{\mathcal{JF}}$ is the unique Kripke-Kleene model of $\mathcal{JF}$.*
- *The partial stable fixpoints of $A_{\mathcal{JF}}$ are exactly the stable models of $\mathcal{JF}$.*
- *The well-founded fixpoint of $A_{\mathcal{JF}}$ is the unique well-founded model of $\mathcal{JF}$.*

These four points are proven independently; the first follows directly from our observation that $A_{\mathcal{JF}}$ and $\mathcal{S}^{\mathcal{B}_{\mathrm{sp}}}_{\mathcal{JF}}$ are in fact the same operator.

Proposition 9. *The partial supported fixpoints of $A_{\mathcal{JF}}$ are exactly the supported models of $\mathcal{JF}$.*

Given the correspondence between supported semantics, the result for Kripke-Kleene semantics follows quite easily.

Proposition 10. *The Kripke-Kleene fixpoint of $A_{\mathcal{JF}}$ is equal to the unique $\mathcal{B}_{\mathrm{KK}}$-model of $\mathcal{JF}$.*

Proof sketch. By combining Propositions 4, and 9, we get that the unique $\mathcal{B}_{\mathrm{KK}}$-model (denoted here $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}$) is a fixpoint of $A_{\mathcal{JF}}$. All that is left to show is that it is the least precise one. Assume towards contradiction that this is not the case, i.e., that there is a fixpoint $\mathcal{I}$ of $A_{\mathcal{JF}}$ such that $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}} \not\leq_p \mathcal{I}$. From this we can find an $x \in \mathcal{F}_d$ such that either $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{f}$ and $\mathcal{I}(x) = \mathbf{t}$, or $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{f}$ and $\mathcal{I}(x) = \mathbf{u}$. In both cases, we have $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{f}$; hence every justification J with x as internal node has a finite branch starting with x mapped to an open fact y with $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(y) = \mathbf{f}$. However, since $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}$ and $\mathcal{I}$ agree on open facts, this also means that $\mathrm{SV}_{\mathcal{B}'_{\mathrm{sp}}}(x, \mathcal{I}) = \mathbf{f}$, contradicting that $\mathcal{I}$ is a $\mathcal{B}'_{\mathrm{sp}}$-model. □

The proof of the third point of Theorem 3 is split in two parts, proven separately in the following propositions.

Proposition 11. *Each stable model of $\mathcal{JF}$ is a partial stable fixpoint of $A_{\mathcal{JF}}$.*

Proof sketch. Take a $\mathcal{B}_{\mathrm{st}}$-model $\mathcal{I} = (I_1, I_2)$ of $\mathcal{JF}$. To show that $\mathcal{I}$ is a partial stable fixpoint, we have two lfp equations to prove. We focus on $\mathrm{lfp}(A_{\mathcal{JF}}(\cdot, I_2)_1) = I_1$. We know from Theorem 1 that a justification J exists that justifies all facts in I_1. Now, this specific justification induces a dependency order on the facts in I_1, defined as $y \preceq_J x$ if y is reachable in J from x through positive facts. Using the definition of stable branch evaluation, we can see that this order is well-founded and subsequenty prove using well-founded induction on this order that all facts in I_1 must be in $\mathrm{lfp}(A_{\mathcal{JF}}(\cdot, I_2)_1)$. □

For the other direction, we first need the following lemma.

Lemma 4. *Let $\mathcal{I}$ be a $\mathcal{B}_{\mathrm{sp}}$-model and $x \in \mathcal{F}_+$ with $\mathrm{SV}_{\mathcal{B}_{\mathrm{sp}}}(x, \mathcal{I}) = \mathbf{f}$. It holds that $\mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, \mathcal{I}) = \mathbf{f}$.*

Proposition 12. *Each partial stable fixpoint of $A_{\mathcal{JF}}$ is a stable model of $\mathcal{JF}$.*

Proof sketch. Let $\mathcal{I} = (I_1, I_2)$ be a partial stable fixpoint of $A_{\mathcal{JF}}$. We prove that $\mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, \mathcal{I}) = \mathcal{I}(x)$ for $x \in \mathcal{F}_+$. By consistency of $\mathcal{S}^{\mathcal{B}_{\mathrm{st}}}_{\mathcal{JF}}$ [Marynissen *et al.*, 2018], this proves that $\mathcal{I}$ is a stable model of $\mathcal{JF}$. The proof consists of three parts.

1. $\mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, \mathcal{I}) = \mathcal{I}(x) = \mathbf{t}$ for all $x \in I_1$.
2. $\mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, \mathcal{I}) = \mathcal{I}(x) = \mathbf{u}$ for all $x \in I_2 \setminus I_1$.
3. $\mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, \mathcal{I}) = \mathcal{I}(x) = \mathbf{f}$ for all $x \in \mathcal{F}_+ \setminus I_2$.

The first part is proven by constructing a possibly non-locally complete justification without infinite branches, every internal node positive, and for every node y we have that $\mathcal{I}(y) = \mathbf{t}$. The construction is possible because I_1 is the least fixpoint of $A_{\mathcal{JF}}(\cdot, I_2)_1$. The second part is proven similarly. Last part is a consequence of Lemma 4. □

Example 4. *Let $\mathcal{F} = \{x, {\sim}x, y, {\sim}y, z, {\sim}z\} \cup \mathcal{L}$ and let R be the complementation of $\{x \leftarrow y; y \leftarrow {\sim}z; z \leftarrow {\sim}x, {\sim}y;\}$, i.e., it adds the rules ${\sim}x \leftarrow {\sim}y$, ${\sim}y \leftarrow z$, ${\sim}z \leftarrow x$ and ${\sim}z \leftarrow y$. The approximator $A_{\mathcal{JF}}$ has three partial stable fixpoints: $(\{x, y\}, \{x, y\})$, $(\{z\}, \{z\})$ and $(\emptyset, \{x, y, z\})$.*

Let us take a look at the fixpoint $(\{x, y\}, \{x, y\})$. Since it is a stable fixpoint, we know that $(\{x, y\}, \{x, y\})$ is a least fixpoint of $A_{\mathcal{JF}}(\cdot, \{x, y\})$. This operator is monotone with respect to $\subseteq$; hence we can construct the fixpoint by iteratively applying the operator on $(\emptyset, \{x, y\})$. This produces the following sequence.

$$(\emptyset, \{x, y\}) \to (\{y\}, \{x, y\}) \to (\{x, y\}, \{x, y\})$$

The first uses the rule $y \leftarrow {\sim}z$, while the second uses $x \leftarrow y$. Combining the two we get the justification $x \to y \to {\sim}z$, which has only true nodes in the model $(\{x, y\}, \{x, y\})$, only positive internal nodes and every defined leaf is negative. This illustrates the first step of the proof of Proposition 12. By extending the found justification, we get a locally complete justification with the same value as the supported value.

The proof of the fourth point of Theorem 3 follows directly from the third point and Proposition 7.

5 Ultimate Semantics for Justification Frames

When applying AFT to new domains, there is not always a clear choice of which approximator to use; the operator on the other hand is often more clear. Denecker *et al.* [2004] studied the space of approximators and observed that consistent approximators can naturally be ordered according to their precision, where more precise approximator also yield more precise results (e.g., if approximator A is more precise than B, then the A-well-founded fixpoint is guaranteed to be more precise than B's). They also observed that the space of approximators of O has a most precise element, called the *ultimate approximator*, denoted $\mathrm{U}(O)$.

In the context of JT, the justification frame uniquely determines the approximator at hand. Still, we show that it is possible to obtain ultimate semantics here as well. To do so, we will develop a method to transform a justification frame $\mathcal{JF}$ into its ultimate frame $\mathrm{U}(\mathcal{JF})$. We will then show that the approximator associated to $\mathrm{U}(\mathcal{JF})$ is indeed the ultimate approximator of $O_{\mathcal{JF}}$. The result is a generic mechanism to go from any semantics induced by JT (for arbitrary branch evaluations – not just for those that have an AFT counterpart) to an ultimate variant thereof. Our construction is as follows:

Definition 14. Let $\mathcal{JF}$ be a complementary justification frame. Let X be the set rules with a positive head. Let X^* be the least (w.r.t. $\subseteq$ of R) set containing X that is closed under the addition of rules $x \leftarrow A$ if there is a rule $x \leftarrow B$ with $B \subseteq A$, or if there are rules $x \leftarrow \{y\} \cup A$, $x \leftarrow \{{\sim}y\} \cup A$. Let Y be the complementation of X. Then $\mathrm{U}(\mathcal{JF})$ is defined to be the complementary justification frame $\langle \mathcal{F}, \mathcal{F}_d, Y \rangle$.

Example 5. *Let $\mathcal{F}_d = \{x, {\sim}x\}$ and $\mathcal{F}_o = \{a, {\sim}a, b, {\sim}b\} \cup \mathcal{F}_o$. Take R to be the complementation of $\{x \leftarrow a, x \; x \leftarrow b, {\sim}x\}$, which adds the following rules*

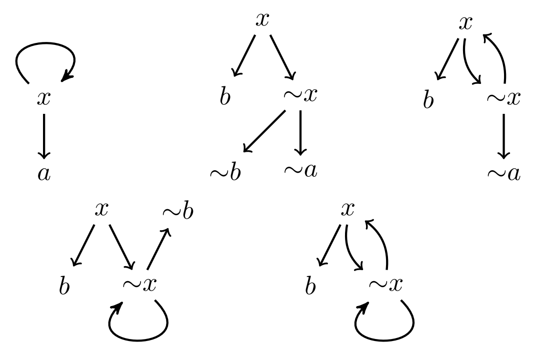
$${\sim}x \leftarrow {\sim}a, {\sim}b \qquad {\sim}x \leftarrow {\sim}a, x$$
$${\sim}x \leftarrow b, {\sim}x \qquad {\sim}x \leftarrow x, {\sim}x$$

A rule $x \leftarrow A$ is minimal, if there is no rule $x \leftarrow B$ with $B \subset A$. For determining the supported value, you only need *to take minimal rules into account. The justification frame* $\mathrm{U}(\mathcal{JF})$ *has exactly the following minimal rules:*

$$x \leftarrow a, x \qquad x \leftarrow b, \sim x \qquad x \leftarrow a, b$$
$$\sim x \leftarrow \sim a, \sim b \qquad \sim x \leftarrow \sim a, x \qquad \sim x \leftarrow \sim b, \sim x$$

Of course, it contains many non-minimal rules, for example $x \leftarrow a, b, x$.

The justifications in the original system containing x *as an internal nodes are exactly the following:*



Assume from now we are working under $\mathcal{B}_{\mathrm{st}}$*. The value of the upper left justification for* x *is* $\mathbf{f}$ *in every interpretation. The values of the other justifications for* x *are at most* $\mathbf{u}$ *in* $\mathcal{B}_{\mathrm{st}}$*-models. If it would be* $\mathbf{t}$*, then the value of these justifications for* x *is equal to the value of* $\sim x$*, which is* $\mathbf{f}$.

By taking the ultimate justification frame, the rule $x \leftarrow a, b$ *is added and the complementary rule* $\sim x \leftarrow x, \sim x$ *is removed. This allows for the justification* $a \leftarrow x \rightarrow b$*. If the interpretation of* a *and* b *is* $\mathbf{t}$*, then the value of this justification for* x *is* $\mathbf{t}$*. Therefore,* $(\{a, b, x\}, \{a, b, x\})$ *is an ultimate stable model, while not a stable model. Note that the lower right justification is not a justification in* $\mathrm{U}(\mathcal{JF})$*. If it would be, then this is a true justification for* $\sim x$ *contradicting the consistency.*

It can be seen that the construction adds rules to $\mathcal{JF}$ in two cases. For the first type, if $x \leftarrow B$ is a rule in R with $B \subseteq A$, then if B is sufficient to derive x, clearly so is A. The second type of rule addition essentially performs some sort of case splitting. It states that if a set of facts A can be used with either y or $\sim y$ to derive x, then the essence for deriving x is the set A itself. In that case, the rule $x \leftarrow A$ is added to the ultimate frame. It turns out that this rule of case splitting is indeed sufficient to reconstruct the ultimate semantics in JT. This is formalized in the main theorem of this section:

Theorem 4. *For any frame* $\mathcal{JF}$*,* $A_{\mathrm{U}(\mathcal{JF})} = \mathrm{U}(O_{\mathcal{JF}})$.

An immediate corollary is, for instance that the set of stable models of $\mathrm{U}(\mathcal{JF})$ equals the set of ultimate stable fixpoints of $O_{\mathcal{JF}}$, and similarly for other semantics. Recall that, in the context of lattices with the subset order, which is what we are concerned with here, the ultimate approximator is defined as follows [Denecker *et al.*, 2004]:

$$\mathrm{U}(O)(I_1, I_2) = \left(\bigcap_{I_1 \subseteq K \subseteq I_2} O(K), \bigcup_{I_1 \subseteq K \subseteq I_2} O(K) \right). \quad (1)$$

The proof of Theorem 4 makes use of the following intermediate results.

Lemma 5. *Let* $\mathcal{I}$ *be an interpretation and* $x \in \mathcal{F}_d$.

If $O_{\mathcal{JF}}(\mathcal{I}')(x) = \mathbf{t}$ *(respectively* $\mathbf{f}$*) for all exact interpretations* $\mathcal{I}'$ *with* $\mathcal{I}' \geq_p \mathcal{I}$*, then* $A_{\mathrm{U}(\mathcal{JF})}(\mathcal{I})(x) = \mathbf{t}$ *(resp.* $\mathbf{f}$*).*

Proof sketch. Take $X = \{y \in \mathcal{F}_d \mid \mathcal{I}(x) = \mathbf{u}\}$ and let $\mathcal{I} = (I_1, I_2)$. We prove for all $Y \subseteq X$ and all complete consistent sets A over $X \setminus Y$ that $x \leftarrow \{\mathbf{t}\} \cup I_1 \cup \sim(\mathcal{F}_+ \setminus I_2) \cup A$ is a rule in $\mathrm{U}(\mathcal{JF})$. If $Y = X$, then we get that $x \leftarrow \{\mathbf{t}\} \cup I_1 \cup \sim(\mathcal{F}_+ \setminus I_2)$ is a rule in $\mathrm{U}(\mathcal{JF})$ such that its body is true in $\mathcal{I}$, which completes the proof. Our claim is proved by transfinite induction on the size of Y. □

Combining this lemma with Eq. (1) of the ultimate approximator immediately yields that the operator $A_{\mathrm{U}(\mathcal{JF})}(\mathcal{I})$ is as least as precise as the ultimate approximator of $O_{\mathcal{JF}}$.

Lemma 6. *For all* $\mathcal{I}$ *we have* $\mathrm{U}(O_{\mathcal{JF}})(\mathcal{I}) \leq_p A_{\mathrm{U}(\mathcal{JF})}(\mathcal{I})$.

Since the ultimate approximator is the most precise approximator of any given operator, all that is left to prove, to indeed obtain Theorem 4 is that $A_{\mathrm{U}(\mathcal{JF})}$ indeed approximates $O_{\mathcal{JF}}$. That is the content of the last lemma.

Lemma 7. $A_{\mathrm{U}(\mathcal{JF})}$ *is an approximator of* $O_{\mathcal{JF}}$.

6 Conclusion

In this paper, we presented a general mechanism to translate justification frames into approximating operators and showed that this transformation preserves all semantics the two formalisms have in common. The correspondence we established provides ample opportunity for future work and in fact probably generates more questions than it answers.

By embedding JT in AFT, JT gets access to a rich body of theoretical results developed for AFT, but of course said results are only directly applicable to branch evaluations that have a counterpart in AFT. A question that immediately arises is whether results such as stratification results also apply to other branch evaluations, and which assumptions on branch evaluations would be required for that. Another question that pops up on the JT side is whether concepts such as *groundedness* [Bogaerts *et al.*, 2015] can be transferred.

On the AFT side, this embedding calls for a general algebraic study of *explanations*. Indeed, for certain approximators, namely those that "come from" a justification frame, our results give us a method for answering certain *why* questions in a graph-based manner (with justifications). Lifting this notion of explanation to general approximators would benefit domains of logics that are covered by AFT but not by JT, such as auto-epistemic logic [Moore, 1985] and default logic [Reiter, 1980].

A last question that emerges naturally is how nesting of justification frames, as defined by Denecker *et al.* [2015] fits into this story, and whether it can give rise to notions of nested operators on the AFT side.

Acknowledgements

Many thanks to Maurice Bruynooghe for the valuable comments and suggestions on draft versions. This research received partial funding from the FWO Flanders project G0B2221N.

References

[Bi *et al.*, 2014] Yi Bi, Jia-Huai You, and Zhiyong Feng. A generalization of approximation fixpoint theory and application. In *Proceedings of RR*, pages 45–59, 2014.

[Bogaerts and Cruz-Filipe, 2018] Bart Bogaerts and Luís Cruz-Filipe. Fixpoint semantics for active integrity constraints. *AIJ*, 255:43–70, 2018.

[Bogaerts and Cruz-Filipe, 2021] Bart Bogaerts and Luís Cruz-Filipe. Stratification in approximation fixpoint theory and its application to active integrity constraints. *ACM Trans. Comput. Logic*, 22(1), January 2021.

[Bogaerts and Van den Broeck, 2015] Bart Bogaerts and Guy Van den Broeck. Knowledge compilation of logic programs using approximation fixpoint theory. *TPLP*, 15(4–5):464–480, 2015.

[Bogaerts and Weinzierl, 2018] Bart Bogaerts and Antonius Weinzierl. Exploiting justifications for lazy grounding of answer set programs. In *Proceedings of IJCAI*, pages 1737–1745, 2018.

[Bogaerts *et al.*, 2015] Bart Bogaerts, Joost Vennekens, and Marc Denecker. Grounded fixpoints and their applications in knowledge representation. *AIJ*, 224:51–71, 2015.

[Cabalar *et al.*, 2014] Pedro Cabalar, Jorge Fandinno, and Michael Fink. Causal graph justifications of logic programs. *TPLP*, 14(4–5):603–618, 2014.

[Charalambidis *et al.*, 2018] Angelos Charalambidis, Panos Rondogiannis, and Ioanna Symeonidou. Approximation fixpoint theory and the well-founded semantics of higher-order logic programs. *TPLP*, 18(3-4):421–437, 2018.

[Damásio *et al.*, 2013] Carlos Viegas Damásio, Anastasia Analyti, and Grigoris Antoniou. Justifications for logic programming. In *Proceedings of LPNMR*, pages 530–542, 2013.

[Denecker and De Schreye, 1993] Marc Denecker and Danny De Schreye. Justification semantics: A unifying framework for the semantics of logic programs. In *Proceedings of LPNMR*, pages 365–379, 1993.

[Denecker *et al.*, 2000] Marc Denecker, Victor Marek, and Mirosław Truszczyński. Approximations, stable operators, well-founded fixpoints and applications in nonmonotonic reasoning. In *Logic-Based Artificial Intelligence*, pages 127–144, 2000.

[Denecker *et al.*, 2003] Marc Denecker, Victor Marek, and Mirosław Truszczyński. Uniform semantic treatment of default and autoepistemic logics. *AIJ*, 143(1):79–122, 2003.

[Denecker *et al.*, 2004] 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.

[Denecker *et al.*, 2015] Marc Denecker, Gerhard Brewka, and Hannes Strass. A formal theory of justifications. In *Proceedings of LPNMR*, pages 250–264, 2015.

[Fages, 1990] F. Fages. A New Fixpoint Semantis for General Logic Programs Compared with the Well-Founded and the Stable Model Semantics. In *ICLP*, page 443. MIT Press, 1990.

[Fitting, 2002] Melvin Fitting. Fixpoint semantics for logic programming — A survey. *Theoretical Computer Science*, 278(1-2):25–51, 2002.

[Gebser *et al.*, 2009] Martin Gebser, Roland Kaminski, Benjamin Kaufmann, and Torsten Schaub. On the implementation of weight constraint rules in conflict-driven ASP solvers. In *ICLP*, pages 250–264, 2009.

[Lapauw *et al.*, 2020] Ruben Lapauw, Maurice Bruynooghe, and Marc Denecker. Improving parity game solvers with justifications. In *Proceedings of VMCAI*, pages 449–470, 2020.

[Marynissen *et al.*, 2018] Simon Marynissen, Niko Passchyn, Bart Bogaerts, and Marc Denecker. Consistency in justification theory. In *Proceedings of NMR*, pages 41–52, 2018.

[Marynissen *et al.*, 2020] Simon Marynissen, Bart Bogaerts, and Marc Denecker. Exploiting game theory for analysing justifications. *Theory Pract. Log. Program.*, 20(6):880–894, 2020.

[Moore, 1985] Robert C. Moore. Semantical considerations on nonmonotonic logic. *AIJ*, 25(1):75–94, 1985.

[Passchyn, 2017] Niko Passchyn. Justification frames. introducing splittable branch evaluations, 2017. Master Thesis; Denecker, Marc (supervisor).

[Reiter, 1980] Raymond Reiter. A logic for default reasoning. *AIJ*, 13(1-2):81–132, 1980.

[Schulz and Toni, 2013] Claudia Schulz and Francesca Toni. ABA-based answer set justification. *TPLP*, 13(4–5-Online-Supplement), 2013.

[Strass, 2013] Hannes Strass. Approximating operators and semantics for abstract dialectical frameworks. *AIJ*, 205:39–70, 2013.

[Tarski, 1955] Alfred Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 1955.

[Vennekens *et al.*, 2006] Joost Vennekens, David Gilis, and Marc Denecker. Splitting an operator: Algebraic modularity results for logics with fixpoint semantics. *ACM Trans. Comput. Log.*, 7(4):765–797, 2006.

[Vennekens *et al.*, 2007] Joost Vennekens, Maarten Mariën, Johan Wittocx, and Marc Denecker. Predicate introduction for logics with a fixpoint semantics. Parts I and II. *Fundamenta Informaticae*, 79(1-2):187–227, 2007.

A Full proofs

Lemma 1. Take $\mathcal{JS} = \langle \mathcal{F}, \mathcal{F}_d, R, \mathcal{B} \rangle$ with $\mathcal{B}$ consistent. Every interpretation $\mathcal{I}$ such that $\mathrm{SV}_{\mathcal{JS}}(x, \mathcal{I}) \geq_t \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$, is a $\mathcal{B}$-model of $\mathcal{JF}$.

Proof. For all $x \in \mathcal{F}_d$ we have that $\mathrm{SV}_{\mathcal{JS}}(\sim x, \mathcal{I}) \geq_t \mathcal{I}(\sim x)$. Marynissen *et al.* [2020] proved that if $\mathcal{B}$ is consistent, then $\mathrm{SV}_{\mathcal{JS}}(\sim x, \mathcal{I}) \leq_t \sim \mathrm{SV}_{\mathcal{JS}}(x, \mathcal{I})$. Therefore, $\mathcal{I}(\sim x) \leq_t \sim \mathrm{SV}_{\mathcal{JS}}(x, \mathcal{I})$, or that $\mathrm{SV}_{\mathcal{JS}}(x, \mathcal{I}) \leq_t \sim \mathcal{I}(\sim x) = \mathcal{I}(x)$. This completes the proof that $\mathrm{SV}_{\mathcal{JS}}(x, \mathcal{I}) = \mathcal{I}(x)$, i.e., $\mathcal{I}$ is a $\mathcal{B}$-model of $\mathcal{JF}$. □

Proposition 1. The two supported branch evaluations $\mathcal{B}_{\mathrm{sp}}$ and $\mathcal{B}'_{\mathrm{sp}}$ are equivalent.

Proof. Remark: In this proof, we make use of Theorem 1, which, for presentational purposes is stated later in the paper. However, it can be verified that Theorem 1 is proven completely independent from the current proposition.

Take a $\mathcal{B}_{\mathrm{sp}}$-model $\mathcal{I}$. By Theorem 1, there is a justification J such that $\mathrm{val}_{\mathcal{B}_{\mathrm{sp}}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}_{\mathrm{sp}}}(x, \mathcal{I}) = \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$. For any rule $x \leftarrow D$ in J, we have that $\mathcal{I}(x) \leq_t \mathcal{I}(z)$ for all $z \in D$. Therefore, by iteratively applying this result, it holds that for all leafs z in J reachable from x, we have that $\mathcal{I}(x) \leq_t \mathcal{I}(z)$. This means that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(x, J, \mathcal{I}) \geq_t \mathrm{val}_{\mathcal{B}_{\mathrm{sp}}}(x, J, \mathcal{I}) = \mathcal{I}(x)$, which implies that $\mathrm{SV}_{\mathcal{B}'_{\mathrm{sp}}}(x, \mathcal{I}) \geq_t \mathcal{I}(x)$. This holds for all $x \in \mathcal{F}_d$; hence by Lemma 1 we have that $\mathcal{I}$ is a $\mathcal{B}'_{\mathrm{sp}}$-model.

Take a $\mathcal{B}'_{\mathrm{sp}}$-model $\mathcal{I}$. By Theorem 1, there is a justification J such that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}'_{\mathrm{sp}}}(x, \mathcal{I}) = \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$. We prove that for every child y of x in J we have that $\mathcal{I}(x) \leq_t \mathcal{I}(y)$. Let y be a child of x so that there is an infinite J-branch which starts with $x \to y$. Then by definition of $\mathcal{B}'_{\mathrm{sp}}$ we have that $\mathcal{I}(x) = \mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(x, J, \mathcal{I}) \leq_t \mathcal{I}(y)$. Let y be a child of x so that there is no infinite J-branch starting with $x \to y$. This means that $\mathcal{I}(y) = \mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(y, J, \mathcal{I}) \geq_t \mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(x, J, \mathcal{I}) = \mathcal{I}(x)$.

This implies that $\mathrm{val}_{\mathcal{B}_{\mathrm{sp}}}(x, J, \mathcal{I}) \geq_t \mathcal{I}(x)$. Therefore, $\mathrm{SV}_{\mathcal{B}_{\mathrm{sp}}}(x, \mathcal{I}) \geq_t \mathcal{I}(x)$. This holds for all $x \in \mathcal{F}_d$; hence $\mathcal{I}$ is a $\mathcal{B}_{\mathrm{sp}}$-model by Lemma 1. □

Proposition 2. The two stable branch evaluations $\mathcal{B}_{\mathrm{st}}$ and $\mathcal{B}'_{\mathrm{st}}$ are equivalent.

Proof. Take a $\mathcal{B}_{\mathrm{st}}$-model $\mathcal{I}$. By Theorem 1, there is a justification J such that $\mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, \mathcal{I}) = \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$. For any finite branch $\mathbf{b}$ in J starting with x, we have for the first sign switch y, that $\mathcal{I}(x) \leq_t \mathcal{I}(y)$ since $\mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x, J, \mathcal{I}) = \mathcal{I}(x)$. Therefore, by applying induction, we have that $\mathcal{I}(x) \leq_t \mathcal{I}(\text{last element of } \mathbf{b})$. This means that J is a justification for x with $\mathrm{val}_{\mathcal{B}'_{\mathrm{st}}}(x, J, \mathcal{I}) \geq_t \mathcal{I}(x)$; hence $\mathrm{SV}_{\mathcal{B}'_{\mathrm{st}}}(x, \mathcal{I}) \geq_t \mathcal{I}(x)$. Then $\mathcal{I}$ is a $\mathcal{B}'_{\mathrm{st}}$-model by Lemma 1.

Take a $\mathcal{B}'_{\mathrm{st}}$-model $\mathcal{I}$. By Theorem 1, there is a justification J such that $\mathrm{val}_{\mathcal{B}'_{\mathrm{st}}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}'_{\mathrm{st}}}(x, \mathcal{I}) = \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$. Take a finite branch $\mathbf{b}$ with a first sign switch y. We prove that $\mathcal{I}(y) \geq_t \mathcal{I}(x)$ since that would prove that $\mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x, J, \mathcal{I}) \geq_t \mathcal{I}(x)$, and by Lemma 1 this proves that $\mathcal{I}$ is $\mathcal{B}_{\mathrm{st}}$-model. If y is the first sign switch of an infinite branch $\mathbf{b}^*$ starting with x, then $\mathcal{I}(x) = \mathrm{val}_{\mathcal{B}'_{\mathrm{st}}}(x, J, \mathcal{I}) \leq_t \mathcal{I}(\mathcal{B}'_{\mathrm{st}}(\mathbf{b}^*)) = \mathcal{I}(y)$. So we can assume that this is not the case. It suffices to prove that $\mathcal{I}(\mathcal{B}'_{\mathrm{st}}(\mathbf{b}^*)) \geq_t \mathcal{I}(x)$ for every $\mathbf{b}^*$ in $B_J(y)$ since that implies that $\mathcal{I}(y) = \mathrm{val}_{\mathcal{B}'_{\mathrm{st}}}(y, J, \mathcal{I}) \geq_t \mathcal{I}(x)$. The branch $\mathbf{b}^*$ cannot be infinite since then the infinite J-branch $x \to \cdots \to \mathbf{b}^*$ has y as first sign switch. If $\mathbf{b}^*$ is finite, then there is a finite branch $\mathbf{b}'$ in $B_J(x)$ so that $\mathcal{B}'_{\mathrm{st}}(\mathbf{b}^*) = \mathcal{B}'_{\mathrm{st}}(\mathbf{b}')$ and thus $\mathcal{I}(\mathcal{B}'_{\mathrm{st}}(\mathbf{b}^*)) \geq_t \mathrm{val}_{\mathcal{B}'_{\mathrm{st}}}(x, J, \mathcal{I}) = \mathcal{I}(x)$. □

Proposition 3. If $\mathcal{JF}$ is a justification frame and $\mathcal{B}$ a parametric branch evaluation, then $\mathcal{JF}$ has a single $\mathcal{B}$-model.

Proof. The value of a justification only depends on the interpretation of the open facts; hence $\mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}) = \mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}')$ for every two interpretations $\mathcal{I}$ and $\mathcal{I}$ that agree on the open facts. This means that $\mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}')$. The unique $\mathcal{B}$-model of $\mathcal{JF}$ is then equal to $\mathcal{S}^{\mathcal{B}}_{\mathcal{JF}}(\mathcal{I})$ for any interpretation $\mathcal{I}$. □

Proposition 4. The unique $\mathcal{B}_{\mathrm{KK}}$-model is a $\mathcal{B}_{\mathrm{sp}}$-model.

Proof. Let $\mathcal{I}$ be the unique $\mathcal{B}_{\mathrm{KK}}$-model of $\mathcal{JF}$. By Theorem 1, there is a justification J such that $\mathrm{val}_{\mathcal{B}_{\mathrm{KK}}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}_{\mathrm{KK}}}(x, \mathcal{I}) = \mathcal{I}(x)$. By Lemma 1, it suffices to prove that $\mathrm{val}_{\mathcal{B}_{\mathrm{sp}}}(x, J, \mathcal{I}) \geq_t \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$.

Since $\mathcal{B}_{\mathrm{KK}}$ is transitive, we have for every node y reachable from x in J that $\mathcal{I}(x) = \mathrm{val}_{\mathcal{B}_{\mathrm{KK}}}(x, J, \mathcal{I}) \leq_t \mathrm{val}_{\mathcal{B}_{\mathrm{KK}}}(y, J, \mathcal{I}) = \mathcal{I}(y)$. The branch evaluation $\mathcal{B}_{\mathrm{sp}}$ always maps to a node reachable from the start node. Therefore, $\mathrm{val}_{\mathcal{B}_{\mathrm{sp}}}(x, J, \mathcal{I}) \geq_t \mathcal{I}(x)$, which concludes the proof. □

Proposition 5. The well-founded model of $\mathcal{JF}$ is a stable model of $\mathcal{JF}$.

Proof. Let $\mathcal{I}$ be the unique well-founded model of $\mathcal{JF}$. By Theorem 1, there is a justification J such that $\mathrm{val}_{\mathcal{B}_{\mathrm{wf}}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x, \mathcal{I}) = \mathcal{I}(x)$. By Lemma 1, it suffices to prove that $\mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, J, \mathcal{I}) \geq_t \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$.

For any internal node y reachable from x in J we have that $\mathcal{I}(x) \leq_t \mathcal{I}(y)$. Indeed, since $\mathcal{B}_{\mathrm{wf}}$ is transitive, we have that for every $\mathbf{b} \in B_J(y)$ there is a branch $\mathbf{b}'$ in $B_J(x)$ so that $\mathcal{B}(\mathbf{b}) = \mathcal{B}(\mathbf{b}')$. This means that $\mathcal{I}(y) = \mathrm{val}_{\mathcal{B}_{\mathrm{wf}}}(y, J, \mathcal{I}) \geq_t \mathrm{val}_{\mathcal{B}_{\mathrm{wf}}}(x, J, \mathcal{I}) = \mathcal{I}(x)$.

For every $\mathbf{b}$ in $B_J(x)$ we have that $\mathcal{B}_{\mathrm{st}}(\mathbf{b})$ is mapped to an element in $\mathbf{b}$ or that $\mathcal{B}_{\mathrm{st}}(\mathbf{b}) = \mathcal{B}_{\mathrm{wf}}(\mathbf{b})$. In both cases, we know that $\mathcal{I}(x) \leq_t \mathcal{I}(\mathcal{B}_{\mathrm{st}}(\mathbf{b}))$. This means that $\mathcal{I}(x) \leq_t \mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x, J, \mathcal{I}) \leq_t \mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, \mathcal{I})$. This concludes the proof that $\mathcal{I}$ is a stable model of $\mathcal{JF}$. □

Proposition 6. Every stable justification model is a supported justification model.

Proof. Take a $\mathcal{B}_{\mathrm{st}}$-model $\mathcal{I}$. By Theorem 1, there is a justification J such that $\mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x, \mathcal{I}) = \mathcal{I}(x)$. Let y be a child of x in J. We prove that $\mathcal{I}(x) \leq_t \mathcal{I}(y)$. If $y \in \mathcal{F}_o$, then by definition of J, we have that $\mathcal{I}(x) \leq_t \mathcal{I}(y)$. If $y \in \mathcal{F}_d$, there are two possibilities: If y has a different sign than x, then by definition of J, we have $\mathcal{I}(x) \leq_t \mathcal{I}(y)$. If y has the same sign as x, then for every $\mathbf{b} \in B_J(y)$ we have that $x \to \mathbf{b}$ is in $B_J(x)$ with $\mathcal{B}(\mathbf{b}) = \mathcal{B}(x \to \mathbf{b})$. Therefore, $\mathcal{I}(y) = \mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(y, J, \mathcal{I}) \geq_t \mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x, J, \mathcal{I}) = \mathcal{I}(x)$. Therefore, we have proved that $\mathrm{val}_{\mathcal{B}_{\mathrm{sp}}}(x, J, \mathcal{I}) \geq_t \mathcal{I}(x)$; hence

$\mathrm{SV}_{\mathcal{B}_{\mathrm{sp}}}(x,\mathcal{I}) \geq_t \mathcal{I}(x)$. Then, by Lemma 1, $\mathcal{I}$ is a $\mathcal{B}_{\mathrm{sp}}$-model. □

Lemma 2. Let $\mathcal{I}$ be a stable justification model. If $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) = \mathbf{f}$, then $\mathcal{I}(x) = \mathbf{f}$. If $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) = \mathbf{t}$, then $\mathcal{I}(x) = \mathbf{t}$.

Proof. For the first, it suffices to prove that $\mathcal{I}(x) \geq_t \mathbf{u}$ implies $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) \geq_t \mathbf{u}$. By Theorem 1, there is a justification J such that $\mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(y,J,\mathcal{I}) = \mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(y,\mathcal{I}) = \mathcal{I}(y)$ for all defined facts y.

So take a defined x with $\mathcal{I}(x) \geq_t \mathbf{u}$. The only way that we have $\mathrm{val}_{\mathcal{B}_{\mathrm{wf}}}(x,J,\mathcal{I}) = \mathbf{f}$ is when $B_J(x)$ contains a branch $\mathbf{b}$ with a positive tail. Let $y_1,\dots,y_n$ be the sign switches in $\mathbf{b}$ and $y_0 = x$. It is easy to see that $\mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(y_n,J,\mathcal{I}) = \mathbf{f}$ since $B_J(y_n)$ contains a positive branch. For i with $0 \leq i < n$ we have that $\mathcal{I}(y_i) = \mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(y_i,\mathcal{I}) = \mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(y_i,J,\mathcal{I}) \leq_t \mathcal{I}(y_{i+1})$. Therefore, we obtained that $\mathcal{I}(x) = \mathbf{f}$, which is a contradiction. Therefore, we have that $\mathcal{I}(x) \geq_t \mathbf{u}$ implies that $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) \geq_t \mathbf{u}$.

The second statement follows by the consistency of $\mathcal{B}_{\mathrm{wf}}$. □

Proposition 7. The well-founded model is the $\leq_p$-least stable model.

Proof. Let $\mathcal{I}_{\mathcal{B}_{\mathrm{wf}}}$ be the well-founded model and let $\mathcal{I}$ be any stable model. Lemma 2 and the consistency of the well-founded semantics imply that if $\mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) = \mathbf{f}$, then $\mathcal{I}(x) = \mathbf{f}$. Therefore, $\mathcal{I}_{\mathcal{B}_{\mathrm{wf}}}(x) = \mathrm{SV}_{\mathcal{B}_{\mathrm{wf}}}(x,\mathcal{I}) \leq_p \mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x,\mathcal{I}) = \mathcal{I}(x)$. Hence, $\mathcal{I}_{\mathcal{B}_{\mathrm{wf}}} \leq_p \mathcal{I}$ for all stable models $\mathcal{I}$. Then Proposition 5 finishes the proof. □

Proposition 8. If no rule body in $\mathcal{JF}$ contains $\mathbf{u}$, then $A_{\mathcal{JF}}$ is a consistent approximator of $O_{\mathcal{JF}}$.

Proof. It is clear that $A_{\mathcal{JF}}$ coincides with $O_{\mathcal{JF}}$ on exact interpretations. It suffices to prove that $A_{\mathcal{JF}}(\mathcal{I})(x) \neq \mathbf{u}$ if $\mathcal{I}$ is exact. This follows directly from the fact that no rule body in $\mathcal{JF}$ contains $\mathbf{u}$. □

Lemma 3. For a complementary justification frame $\mathcal{JF}$, the function $A_{\mathcal{JF}}$ and the support operator $\mathcal{S}_{\mathcal{JF}}^{\mathcal{B}_{\mathrm{sp}}}$ are equal.

Proof. Take an interpretation $\mathcal{I}$. For any $x \in \mathcal{F}_+$, it is obvious that $A_{\mathcal{JF}}(\mathcal{I})(x) = \mathcal{S}_{\mathcal{JF}}^{\mathcal{B}_{\mathrm{sp}}}(\mathcal{I})(x)$. Take $x \in \mathcal{F}_-$. We have that $A_{\mathcal{JF}}(\mathcal{I})(x) = {\sim}A_{\mathcal{JF}}(\mathcal{I})({\sim}x) = {\sim}\mathcal{S}_{\mathcal{JF}}^{\mathcal{B}_{\mathrm{sp}}}(\mathcal{I})({\sim}x) = \mathcal{S}_{\mathcal{JF}}^{\mathcal{B}_{\mathrm{sp}}}(\mathcal{I})(x)$, where the last step is by the consistency of $\mathcal{S}_{\mathcal{JF}}^{\mathcal{B}_{\mathrm{sp}}}$ as proved by Marynissen *et al.* [2018]. □

Proposition 9. The partial supported fixpoints of $A_{\mathcal{JF}}$ are exactly the supported models of $\mathcal{JF}$.

Proof. Follows directly from Lemma 3. □

Proposition 10. The Kripke-Kleene fixpoint of $A_{\mathcal{JF}}$ is equal to the unique $\mathcal{B}_{\mathrm{KK}}$-model of $\mathcal{JF}$.

Proof. Let $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}$ be the unique $\mathcal{B}_{\mathrm{KK}}$-model of $\mathcal{JF}$. By Proposition 4, it suffices to prove that $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}$ is the $\leq_p$-least fixpoint of $A_{\mathcal{JF}}$. Assume there is a fixpoint $\mathcal{I}$ of $A_{\mathcal{JF}}$ such that $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}} \not\leq_p \mathcal{I}$. This means there exists an $x \in \mathcal{F}_d$ such that $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) \not\leq_p \mathcal{I}(x)$. There are exactly four cases

1. $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{t}$ and $\mathcal{I}(x) = \mathbf{f}$;
2. $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{f}$ and $\mathcal{I}(x) = \mathbf{t}$;
3. $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{f}$ and $\mathcal{I}(x) = \mathbf{u}$;
4. $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{t}$ and $\mathcal{I}(x) = \mathbf{u}$.

A fact x satisfies the first case if and only if ${\sim}x$ satisfies the second case. The same holds for the third and fourth case. Therefore, it suffices to look at the following two cases:

1. $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{f}$ and $\mathcal{I}(x) = \mathbf{t}$;
2. $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{f}$ and $\mathcal{I}(x) = \mathbf{u}$.

In both cases, we have that $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(x) = \mathbf{f}$; hence every justification J with x as internal node has a finite branch starting with x mapped to an open fact y with $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}(y) = \mathcal{I}(y) = \mathbf{f}$. Therefore, $\mathcal{I}(x) = \mathrm{SV}_{\mathcal{B}'_{\mathrm{sp}}}(x,\mathcal{I}) = \mathbf{f}$. By Proposition 1, $\mathcal{I}$ is also a $\mathcal{B}'_{\mathrm{sp}}$-model; hence $\mathcal{I}(x) = \mathbf{f}$, which contradicts both cases. This completes the proof that $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}} \leq_p \mathcal{I}$; hence $\mathcal{I}_{\mathcal{B}_{\mathrm{KK}}}$ is the $\leq_p$-least fixpoint of $A_{\mathcal{JF}}$. □

Proposition 11. Each stable model of $\mathcal{JF}$ is a partial stable fixpoint of $A_{\mathcal{JF}}$.

Proof. Let $\mathcal{I} = (I_1, I_2)$ be a $\mathcal{B}_{\mathrm{st}}$-model of $\mathcal{JF}$. We prove that $\mathrm{lfp}(A_{\mathcal{JF}}(\cdot, I_2)_1) = I_1$ and that $\mathrm{lfp}(A_{\mathcal{JF}}(I_1,\cdot)_2) = I_2$. By Proposition 6, it holds that $A_{\mathcal{JF}}(\mathcal{I}) = \mathcal{S}_{\mathcal{JF}}^{\mathcal{B}_{\mathrm{sp}}}(\mathcal{I}) = \mathcal{I}$. Therefore, we have that $A_{\mathcal{JF}}(I_1,I_2)_1 = I_1$ and $A_{\mathcal{JF}}(I_1,I_2)_2 = I_2$.

Take $I'_1 \subsetneq I_1$ and assume by contradiction that $A_{\mathcal{JF}}(I'_1,I_2)_1 = I'_1$. Define $\mathcal{I}' = (I'_1, I_2)$. Therefore, $\mathcal{I}' <_p \mathcal{I}$ and $\mathcal{I}(x) = \mathbf{t}$ and $\mathcal{I}'(x) = \mathbf{u}$ for all $x \in I'_1 \setminus I_1$. By Theorem 1, there is a justification J so that $\mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x,J,\mathcal{I}) = \mathrm{SV}_{\mathcal{B}_{\mathrm{st}}}(x,J,\mathcal{I}) = \mathcal{I}(x)$ for all $x \in \mathcal{F}_d$. Define the partial order $\preceq_J$ on I_1: $y \preceq_J x$ if y is reachable in J from x through positive facts. Since J does not contain infinite positive branches starting from a fact $x \in I_1$, we have that $\preceq_J$ does not have infinitely descending chains; hence $\preceq_J$ is well-founded. The set $I_1 \setminus I'_1$ is not empty, hence has a minimal element x with respect to $\preceq_J$. Take a child y of x in J. We prove that $\mathcal{I}'(y) = \mathbf{t}$. If y is open, then $\mathbf{t} = \mathcal{I}(x) = \mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x,J,\mathcal{I}) \leq_t \mathcal{I}(y) = \mathcal{I}'(y)$. If y has a different sign than x, then $\mathbf{t} = \mathcal{I}(x) = \mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x,J,\mathcal{I}) \leq_t \mathcal{I}(y)$. This means that ${\sim}y \notin I_2$ since $y \in \mathcal{F}_-$; hence $\mathcal{I}'(y) = \mathbf{t}$. If y has the same sign as x, then $\mathbf{t} = \mathcal{I}(x) = \mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(x,J,\mathcal{I}) \leq_t \mathrm{val}_{\mathcal{B}_{\mathrm{st}}}(y,J,\mathcal{I}) = \mathcal{I}(y)$. Therefore, $y \in I_1$, which implies that $y \prec_J x$. This means that $y \notin I_1 \setminus I'_1$. We can conclude that $y \in I'_1$; hence $\mathcal{I}'(y) = \mathbf{t}$. This shows that $\mathrm{val}_{\mathcal{B}_{\mathrm{sp}}}(x,J,\mathcal{I}') = \mathbf{t}$, hence $\mathrm{SV}_{\mathcal{B}_{\mathrm{sp}}}(x,\mathcal{I}') = \mathbf{t}$. This implies that $x \in A_{\mathcal{JF}}(I'_1,I_2)_1 = I'_1$, which contradicts that $x \notin I'_1$; hence $I_1 = \mathrm{lfp}(A_{\mathcal{JF}}(\cdot,I_2)_1)$.

Take $I_1 \subseteq I'_2 \subsetneq I_2$ and assume by contradiction that $A_{\mathcal{JF}}(I_1,I'_2)_2 = I'_2$. Define $\mathcal{I}'' = (I_1, I'_2)$. Therefore, $\mathcal{I} <_p \mathcal{I}''$ and $\mathcal{I}(x) = \mathbf{u}$ and $\mathcal{I}''(x) = \mathbf{f}$ for all $x \in I_2 \setminus I'_2$.

Define the partial order $\preceq'_J$ on I_2 the same as before. This order is also well-founded for a similar reason. The set $I_2 \setminus I'_2$ is not empty, hence has a minimal element x with respect to $\preceq'_J$. Take a child y of x in J. We prove that $\mathcal{I}''(y) \geq_t \mathbf{u}$. If y is open, then $\mathbf{u} = \mathcal{I}(x) = \text{val}_{\mathcal{B}_{\text{st}}}(x, J, \mathcal{I}) \leq_t \mathcal{I}(y) = \mathcal{I}''(y)$. If y has a different sign as x, then $\mathbf{u} = \mathcal{I}(x) = \text{val}_{\mathcal{B}_{\text{st}}}(x, J, \mathcal{I}) \leq_t \mathcal{I}(y)$. This means that $\sim y \notin I_2$ or $\sim y \in I_2 \setminus I_1$. Therefore, $\sim y \notin I'_2$ or $\sim y \in I'_2 \setminus I_1$, thus $\mathcal{I}''(y) \geq_t \mathbf{u}$. If y has the same sign as x, then $\mathbf{u} = \mathcal{I}(x) = \text{val}_{\mathcal{B}_{\text{st}}}(x, J, \mathcal{I}) \leq_t \text{val}_{\mathcal{B}_{\text{st}}}(y, J, \mathcal{I}) = \mathcal{I}(y)$. Therefore, $y \in I_2$, which implies that $y \prec'_J x$. This means that $y \notin I_2 \setminus I'_2$, hence $y \in I'_2$. We conclude that $\mathcal{I}''(y) \geq_t \mathbf{u}$. This shows that $\text{val}_{\mathcal{B}_{\text{sp}}}(x, J, \mathcal{I}'') \geq_t \mathbf{u}$, hence $\text{SV}_{\mathcal{B}_{\text{sp}}}(x, \mathcal{I}'') \geq_t \mathbf{u}$. This implies that $x \in A_{\mathcal{JF}}(I_1, I'_2)_2 = I'_2$, which contradicts that $x \notin I'_2$, concluding that $I_2 = \text{lfp}(A_{\mathcal{JF}}(I_1, \cdot)_2)$. This finishes the proof that $\mathcal{I}$ is a partial stable fixpoint of $A_{\mathcal{JF}}$. □

Lemma 4. Let $\mathcal{I}$ be a $\mathcal{B}_{\text{sp}}$-model and $x \in \mathcal{F}_+$ with $\text{SV}_{\mathcal{B}_{\text{sp}}}(x, \mathcal{I}) = \mathbf{f}$. It holds that $\text{SV}_{\mathcal{B}_{\text{st}}}(x, \mathcal{I}) = \mathbf{f}$.

Proof. Take an arbitrary justification J with x as internal node. There is a child y of x with $\mathcal{I}(y) = \mathbf{f}$. If y has a different sign than x, then $\text{val}_{\mathcal{B}_{\text{st}}}(x, J, \mathcal{I}) = \mathbf{f}$. Otherwise, we can construct either a finite branch ending in a z with $\mathcal{I}(z) = \mathbf{f}$ or an infinite branch such that every element z in $\mathbf{b}$ has $\mathcal{I}(z) = \mathbf{f}$ and z has the same sign as x. Since x is positive, this means that $\text{val}_{\mathcal{B}_{\text{st}}}(x, J, \mathcal{I}) = \mathbf{f}$. This concludes the proof that $\text{SV}_{\mathcal{B}_{\text{st}}}(x, \mathcal{I}) = \mathbf{f}$. □

Proposition 12. Each partial stable fixpoint of $A_{\mathcal{JF}}$ is a stable model of $\mathcal{JF}$.

Proof. Let $\mathcal{I} = (I_1, I_2)$ be a partial stable fixpoint of $A_{\mathcal{JF}}$. We prove that $\text{SV}_{\mathcal{B}_{\text{st}}}(x, \mathcal{I}) = \mathcal{I}(x)$ for $x \in \mathcal{F}_+$. By consistency of $\mathcal{S}^{\mathcal{B}_{\text{st}}}_{\mathcal{JF}}$ [Marynissen *et al.*, 2018], this proves that $\mathcal{I}$ is a stable model of $\mathcal{JF}$. We prove our claim in three parts

Part 1: $\text{SV}_{\mathcal{B}_{\text{st}}}(x, \mathcal{I}) = \mathcal{I}(x) = \mathbf{t}$ **for all** $x \in I_1$**.** Since I_1 is $\text{lfp}(A_{\mathcal{JF}}(\cdot, I_2)_1)$, there is a sequence $(K_i)_{i \leq \beta}$ for some ordinal number β so that

- $K_0 = \emptyset$;
- $K_{i+1} = A_{\mathcal{JF}}(K_i, I_2)_1$ for all $i < \beta$;
- $K_\alpha = \bigcup_{i<\alpha} K_i$ for all limit ordinals $\alpha \leq \beta$;
- $K_\beta = I_1$.

Now, for every $x \in I_1$, there is a least ordinal i_x such that $x \notin K_{i_x}$, while $x \in K_{i_x+1}$. This means that there is a rule $x \leftarrow A_x$ such that for all $y \in A_x$ we have that $(K_{i_x}, I_2)(y) = \mathbf{t}$. Since $(K_{i_x}, I_2) \leq_p \mathcal{I}$ we have that $\mathcal{I}(y) = \mathbf{t}$ for all $y \in A_x$.

We now define J to be the justification with exactly the rules $x \leftarrow A_x$ for $x \in I_1$. Every J-branch is finite and ending in an element in $\mathcal{F}_- \cup \mathcal{F}_o$. Indeed, an infinite J-branch $x = x_0 \to x_1 \to \cdots$ implies the existence of strictly decreasing sequence of ordinals $(i_{x_0}, i_{x_1}, \ldots)$. Therefore, the value of any J-branch starting from x is an element of J, hence $\text{val}_{\mathcal{B}_{\text{st}}}(x, J, \mathcal{I}) = \mathbf{t}$.

Part 2: $\text{SV}_{\mathcal{B}_{\text{st}}}(x, \mathcal{I}) = \mathcal{I}(x) = \mathbf{u}$ **for all** $x \in I_2 \setminus I_1$**.** Since I_2 is $\text{lfp}(A_{\mathcal{JF}}(I_1, \cdot)_2)$, there is a sequence $(M_i)_{i \leq \beta}$ for some ordinal number β so that

- $M_0 = I_1$;
- $M_{i+1} = A_{\mathcal{JF}}(I_1, M_i)_2$ for $i < \beta$;
- $M_\alpha = \bigcup_{i<\alpha} M_i$ for limit ordinal $\alpha \leq \beta$;
- $M_\beta = I_2$.

For every $x \in I_2 \setminus I_1$, there is a least ordinal j_x such that $x \notin M_{j_x}$, while $x \in M_{j_x+1}$. Therefore, there is a rule $x \leftarrow C_x$ such that for all $y \in C_x$ we have that $(I_1, M_{j_x})(y) \geq_t \mathbf{u}$. Since $\mathcal{I} \leq_p (I_1, M_{j_x})$ we have that $\mathcal{I}(y) \geq_t \mathbf{u}$ for all $y \in C_x$.

Define J' to be the justification with exactly the rules $x \leftarrow C_x$ for $x \in I_2 \setminus I_1$. By a similar reasoning as in the first part, we have that every J'-branch is finite and ending in $I_1 \cup \mathcal{F}_- \cup \mathcal{F}_o$. Define J^* as $J' \uparrow J$, with J the justification from the first part. A J^*-branch is either a J'-branch or a concatenation of a J'-branch with a J-branch. This means that J^* does not have infinite branches. By construction, we have for every element y in J^* that $\mathcal{I}(y) \geq_t \mathbf{u}$. The evaluation of a J^*-branch is equal to an element in J^*; hence $\text{val}_{\mathcal{B}_{\text{st}}}(x, J^*, \mathcal{I}) \geq_t \mathbf{u}$.

However, every justification for x has a branch $\mathbf{b}$ such that $\mathcal{I}(\mathcal{B}_{\text{st}}(\mathbf{b})) \leq_t \mathbf{u}$; hence $\text{SV}_{\mathcal{B}_{\text{st}}}(x, \mathcal{I}) = \mathbf{u} = \mathcal{I}(x)$. Indeed, for every y with $\mathcal{I}(y) \leq_t \mathbf{u}$ and rule $y \leftarrow C$ there is a $c \in C$ with $\mathcal{I}(x) \leq_t \mathbf{u}$. This constructs a branch $\mathbf{b}$ such that for every element y in $\mathbf{b}$ we have that $\mathcal{I}(y) \leq_t \mathbf{u}$. We know that $\mathcal{B}_{\text{st}}(\mathbf{b}) \neq \mathbf{t}$; otherwise $\mathbf{b}$ is completely negative or ending in $\mathbf{t}$. Therefore, $\mathcal{B}_{\text{st}}$ maps $\mathbf{b}$ to an element in $\mathbf{b}$ or to $\mathbf{f}$ or $\mathbf{u}$. This proves that $\mathcal{I}(\mathcal{B}_{\text{st}}(\mathbf{b})) \leq_t \mathbf{u}$.

Part 3: $\text{SV}_{\mathcal{B}_{\text{st}}}(x, \mathcal{I}) = \mathcal{I}(x) = \mathbf{f}$ **for all** $x \in \mathcal{F}_+ \setminus I_2$**.** This is immediate from Lemma 4. □

Lemma 5. Let $\mathcal{I}$ be an interpretation and $x \in \mathcal{F}_d$.

If $O_{\mathcal{JF}}(\mathcal{I}')(x) = \mathbf{t}$ (respectively $\mathbf{f}$) for all exact interpretations $\mathcal{I}'$ with $\mathcal{I}' \geq_p \mathcal{I}$, then $A_{\text{U}(\mathcal{JF})}(\mathcal{I})(x) = \mathbf{t}$ (resp. $\mathbf{f}$).

Proof. Let $\mathcal{I} = (I_1, I_2)$. We first prove this for the case $\mathbf{t}$. Define $X = \{y \in \mathcal{F}_d \mid \mathcal{I}(x) = \mathbf{u}\}$. We prove for all $Y \subseteq X$ and all complete consistent subsets A over $X \setminus Y$ that the rule $x \leftarrow \{\mathbf{t}\} \cup I_1 \cup \sim(\mathcal{F}_+ \setminus I_2) \cup A$ is a rule in $\text{U}(\mathcal{JF})$. A complete consistent subset A of Z is a subset such that for each $z \in Z$ exactly one of z and $\sim z$ is in A. If $Y = X$, then we get that $x \leftarrow \{\mathbf{t}\} \cup I_1 \cup \sim(\mathcal{F}_+ \setminus I_2)$ is a rule in $\text{U}(\mathcal{JF})$ with every element y in its body we have $\mathcal{I}(y) = \mathbf{t}$, completing our proof. We prove our claim by transfinite induction. Assume $Y = \emptyset$. Take a complete consistent set A over X. Define $K = I_1 \cup (A \cap \mathcal{F}_+)$. Since $\mathcal{I} \leq_p (K, K)$, we have a rule $x \leftarrow B$ in $\mathcal{JF}$ with $(K, K)(b) = \mathbf{t}$ for all $b \in B$. The true facts under B are equal to $\{\mathbf{t}\} \cup I_1 \cup \sim(\mathcal{F}_+ \setminus I_2) \cup A$. Take $b \in B$. Therefore, we can extend this rule to $x \leftarrow I_1 \cup \sim(\mathcal{F}_+ \setminus I_2) \cup A$ in $\text{U}(\mathcal{JF})$.

Take $Y \neq \emptyset$. Assume by inducton the claim holds for all $Y' \subsetneq Y$. Take any $y \in Y$ and a complete consistent set A over $X \setminus Y$. Then $A \cup \{y\}$ and $A \cup \{\sim y\}$ are complete consistent sets over $X \setminus (Y \setminus \{y\})$. So by induction there are rules $x \leftarrow I_1 \cup (\mathcal{F}_+ \setminus I_2) \cup A \cup \{y\}$ and $x \leftarrow I_1 \cup (\mathcal{F}_+ \setminus I_2) \cup A \cup \{y\}$ in $\text{U}(\mathcal{JF})$. This means that $x \leftarrow I_1 \cup (\mathcal{F}_+ \setminus I_2) \cup A$ is a rule in $\text{U}(\mathcal{JF})$.

The case for $\mathbf{f}$ follows easily from the $\mathbf{t}$ case. We have for all exact interpretations $\mathcal{I}'$ with $\mathcal{I} \leq_p \mathcal{I}'$ that $A_{\mathcal{JF}}(\mathcal{I}')(x) = \mathbf{f}$, i.e., by consistency that $A_{\mathcal{JF}}(I, I)(\sim x) = \mathbf{t}$. By Lemma 5, we have that $A_{\text{U}(\mathcal{JF})}(\mathcal{I})(\sim x) = \mathbf{t}$; hence $A_{\text{U}(\mathcal{JF})}(\mathcal{I})(x) = \mathbf{f}$. □

Lemma 6. For all $\mathcal{I}$ we have $\mathrm{U}(O_{\mathcal{JF}})(\mathcal{I}) \leq_p A_{\mathrm{U}(\mathcal{JF})}(\mathcal{I})$.

Proof. Take $x \in \mathcal{F}_+$. If $\mathrm{U}_{\mathcal{JF}}(\mathcal{I})(x) = \mathbf{u}$, then it is obvious that $A_{\mathrm{U}(\mathcal{JF})}(\mathcal{I})(x) \leq_p \mathbf{u} = \mathrm{U}_{\mathcal{JF}}(\mathcal{I})(x)$.

If $\mathrm{U}_{\mathcal{JF}}(\mathcal{I})(x) = \mathbf{t}$, then $x \in \cap_{I_1 \subseteq K \subseteq I_2} A_{\mathcal{JF}}(K,K)_1$; hence $A_{\mathcal{JF}}(K,K)(x) = \mathbf{t}$ for all exact interpretations $(K,K) \geq_p \mathcal{I}$. Therefore, by Lemma 5, we have that $A_{\mathrm{U}(\mathcal{JF})}(\mathcal{I})(x) = \mathbf{t} = \mathrm{U}_{\mathcal{JF}}(\mathcal{I})(x)$.

If $\mathrm{U}_{\mathcal{JF}}(\mathcal{I})(x) = \mathbf{f}$, then $x \notin \cup_{I_1 \subseteq K \subseteq I_2} A_{\mathcal{JF}}(K,K)_1$; hence $A_{\mathcal{JF}}(K,K)(x) = \mathbf{f}$ for all exact interpretations $(K,K) \geq_p \mathcal{I}$. By Lemma 5 it follows that $A_{\mathrm{U}(\mathcal{JF})}(\mathcal{I})(x) = \mathbf{f} = \mathrm{U}_{\mathcal{JF}}(\mathcal{I})(x)$.

Similarly, we can prove for all $x \in \mathcal{F}_-$ that $\mathrm{U}_{\mathcal{JF}}(\mathcal{I})(x) \leq_p A_{\mathrm{U}(\mathcal{JF})}(\mathcal{I})(x)$. ☐

Lemma 7. $A_{\mathrm{U}(\mathcal{JF})}$ is an approximator of $O_{\mathcal{JF}}$.

Proof. Take $I \subseteq \mathcal{F}_+$. Adding a rule $x \leftarrow B$ to $\mathcal{JF}$ if $x \leftarrow A$ is in $\mathcal{JF}$ with $A \subsetneq B$, does not change $A_{\mathcal{JF}}(I,I)$. Similarly, adding a rule $x \leftarrow A$ to $\mathcal{JF}$ if $x \leftarrow A \cup \{y\}$ and $x \leftarrow A \cup \{\sim y\}$ are in $\mathcal{JF}$ does not change $A_{\mathcal{JF}}(I,I)$. This means that $O_{\mathcal{JF}}(I) = A_{\mathcal{JF}}(I,I) = A_{\mathrm{U}(\mathcal{JF})}(I,I)$. ☐

B Pasting justifications

In this section we finish the proofs of Theorems 1 and 2. But before doing that we have to define what it means to past justifications together. Any justification J can be seen as a partial function from $\mathcal{F}_d$ to R. Therefore we can associate a domain to J, denoted $\mathrm{dom}(J)$, which are just the non-leafs of J. We call K an *extension* of J if K coincides with J on the domain of J seen as partial functions.

Definition 15. For any two justifications J and K, the justification $J \uparrow K$ is defined as the justification J on $\mathrm{dom}(J)$ and K on $\mathrm{dom}(K) \setminus \mathrm{dom}(J)$.

It is obvious that $J \uparrow K$ is an extension of J.

The definition of val can be extended to non-locally complete justifications: $\mathrm{val}_{\mathcal{B}}(x,J,\mathcal{I})$ is the minimum of $\mathrm{val}_{\mathcal{B}}(x,K,\mathcal{I})$ for every locally complete extension K of J. This definition is correct since if J is locally complete, then $\mathrm{val}_{\mathcal{B}}(x,J,\mathcal{I}) = \mathrm{val}_{\mathcal{B}}(x,K,\mathcal{I})$ for every locally complete extension K of J.

To prove Theorem 1, we first need to introduce splittable branch evaluations. They were first introduced in the master's thesis of Passchyn [2017].

For a finite branch $\mathbf{b}\colon x_0 \to x_1 \to \cdots \to x_n$, denote $\ell(\mathbf{b}) = n$. If $\mathbf{b}$ is infinite, we say $\ell(\mathbf{b}) = \infty$. Two branches $x_0 \to x_1 \to \cdots$ and $y_0 \to y_1 \to \cdots$ are *identical up to* n if for all $0 \leq i \leq n$, we have $x_i = y_i$. We say a $\mathcal{JF}$-branch $\mathbf{b}\colon x_0 \to x_1 \to \cdots$ is *decided* under $\mathcal{B}$ at $0 < n < \ell(\mathbf{b}) + 1$ if for every $\mathcal{JF}$-branch $\mathbf{b}'$ identical to $\mathbf{b}$ up to n, we have $\mathcal{B}(\mathbf{b}) = \mathcal{B}(\mathbf{b}')$. In this case, we call the $\mathcal{JF}$-path $x_0 \to \cdots \to x_n$ *decided* under $\mathcal{B}$. This means to determine the value of $\mathbf{b}$ under $\mathcal{B}$, we only need the $n+1$ first elements, so all relevant information is located at the beginning of the branch. This allows us to extend $\mathcal{B}$ to paths that are decided. On the other hand, we define $\mathbf{b}$ to be *transitive* under $\mathcal{B}$ at $0 \leq n < \ell(\mathbf{b})$ if

$$\mathcal{B}(\mathbf{b}) = \mathcal{B}(x_n \to x_{n+1} \to \cdots).$$

Intuitively, a branch is transitive if all information needed to evaluate it is located in the tail after n. A branch $\mathbf{b}$ is called *splittable* under $\mathcal{B}$ at $0 \leq n < \ell(\mathbf{b}) + 1$ if it is either decided or transitive under $\mathcal{B}$ at n.

Intuitively, if a branch is splittable at n, then the information to evaluate the branch is either in tail or start, but not in both. If $\mathcal{B}$ is clear from context, 'under $\mathcal{B}$' is left out.

A branch $\mathbf{b}$ is called *first decided* at $0 < i < \ell(\mathbf{b}) + 1$ if $\mathbf{b}$ is decided at $i = 1$ or $\mathbf{b}$ is decided at $i > 1$ and $\mathbf{b}$ is not decided at $i - 1$.

We say that a branch $\mathbf{b}$ is *totally decided* if it is decided at i for every $0 < i < \ell(\mathbf{b}) + 1$. Similarly, $\mathbf{b}$ is *totally transitive* if it is transitive at i for every $0 \leq i < \ell(\mathbf{b})$. A branch $\mathbf{b}$ is *totally splittable* if it is splittable at i for every $0 \leq i < \ell(\mathbf{b}) + 1$.

Definition 16. A branch evaluation $\mathcal{B}$ is called

- *splittable* if every $\mathcal{JF}$-branch is totally splittable;
- *transitive* if every $\mathcal{JF}$-branch is totally transitive;
- *decided* if every $\mathcal{JF}$-branch is totally decided;

for every justification frame $\mathcal{JF}$.

Marynissen *et al.* [2018] proved that any transitive or decided branch evaluation is splittable. We can now go back to proving Theorem 1.

Lemma 8. *Let $\mathcal{B}$ be a splittable branch evaluation, $\mathcal{I}$ an interpretation, $\ell \in \{\mathbf{f},\mathbf{u},\mathbf{t}\}$, and take $x \in \mathcal{F}_d$ with $\mathrm{SV}_{\mathcal{B}}(x,\mathcal{I}) \geq_t \ell$. There is a justification J (not necessarily locally complete) with x as internal node such that $\mathrm{val}_{\mathcal{B}}(y,J,\mathcal{I}) \geq_t \ell$ for all internal nodes y of J.*

Proof. Take a justification K with $\mathrm{val}_{\mathcal{B}}(x,K,\mathcal{I}) = \mathrm{SV}_{\mathcal{B}}(x,\mathcal{I}) \geq_t \ell$. Let D be the internal nodes y in K such that there exists a K-path $x \to \cdots \to y$ that is undecided at y. Note that we have that $x \in D$. Define J to be the restriction of K with domain D. The internal nodes of J are exactly the elements of D. Take $y \in D$ and a locally complete extension M of J. Let $\mathbf{b}\colon y \to y_1 \to y_2 \to \cdots$ be an M-branch. We prove that $\mathrm{val}_{\mathcal{B}}(y,M,\mathcal{I}) \geq_t \ell$ by proving that $\mathcal{I}(\mathbf{b}) \geq_t \ell$ for all $\mathbf{b} \in B_M(y)$. Since M is chosen arbitrary, this means that $\mathrm{val}_{\mathcal{B}}(y,J,\mathcal{I}) \geq_t \ell$.

Assume first that $\mathbf{b}$ is a J-branch, then it is also a K-branch. If $y = x$, then $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \mathrm{val}_{\mathcal{B}}(x,K,\mathcal{I}) \geq_t \ell$. If $y \neq x$, then there is a K-path $x \to \cdots \to y$ that is not decided at y. Hence, $x \to \cdots \to y \to y_1 \to y_2 \to \cdots$ is transitive at y by splittability of $\mathcal{B}$. Therefore, $\mathcal{B}(\mathbf{b}) = \mathcal{B}(x \to \cdots \to y \to y_1 \to y_2 \to \cdots)$. The latter is a K-branch starting with x, thus $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \ell$.

Therefore, we can assume that $\mathbf{b}$ is not a J-branch; hence there is a least i such that y_i is not an internal node of J. This means that y_i is a leaf of J. Since J is a subgraph of K, this means that y_i is either a leaf or an internal node of K. In the former case, y_i is open, thus that $\mathbf{b}$ is a J-branch, which contradicts our assumption. Hence, y_i is an internal node of K. If $y = x$, then y_i is an internal node of K outside of D. Therefore, $\mathbf{b}$ is decided at y_i, so the evaluation of $\mathbf{b}$ is equal to the evaluation of a K-path starting from x. This means that $\mathcal{I}(\mathcal{B}(\mathbf{b})) = \mathcal{I}(\mathcal{B}(\mathbf{b}'))$ for some K-branch $\mathbf{b}'$ starting at x; hence $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \ell$. If $y \neq x$, then there exists a K-branch

$\mathbf{b}'\colon x \to x_1 \to \cdots \to x_m \to y \to y_1 \to y_2 \to \cdots$ that is not decided at y. By splittability of $\mathcal{B}$, $\mathbf{b}'$ is transitive at y: $\mathcal{B}(\mathbf{b}') = \mathcal{B}(\mathbf{b})$. Since $\mathbf{b}'$ is a K-branch starting at x, we have that $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \ell$. □

Definition 17. A justification J is *ℓ-domain supporting* in $\mathcal{I}$ if $\mathrm{val}_{\mathcal{B}}(y, J, \mathcal{I}) \geq_t \ell$ for all internal nodes y of J.

Lemma 9. *Let J and K be two ℓ-domain supporting justifications. The justification $J \uparrow K$ is also ℓ-domain supporting.*

Proof. Take a locally complete extension M of $J \uparrow K$ and an internal node y of $J \uparrow K$. We prove that $\mathrm{val}_{\mathcal{B}}(y, M, \mathcal{I}) \geq_t \ell$. If y is an internal node of J, then $\mathrm{val}_{\mathcal{B}}(y, M, \mathcal{I}) \geq_t \mathrm{val}_{\mathcal{B}}(y, J, \mathcal{I}) \geq_t \ell$ because M is also a locally complete extension of J. So we can assume that y is not an internal node of J; hence y is an internal node of K. We prove for every M-branch $\mathbf{b}$ starting with y that $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \ell$. If $\mathbf{b}$ is a K-branch, then $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \mathrm{val}_{\mathcal{B}}(y, K, \mathcal{I}) \geq_t \ell$. So we can assume that $\mathbf{b}$ is not a K-branch; hence there is a least i such that y_i is not an internal node of K.

Assume first that y_i is an internal node of J. If $\mathbf{b}$ is transitive at i, then $\mathcal{B}(\mathbf{b}) = \mathcal{B}(y_i \to y_{i+1} \to \cdots)$. The latter branch is an M-branch starting with y_i, which is an internal node of J. Hence, by the result above we have that $\mathcal{I}(\mathcal{B}(\mathbf{b})) = \mathcal{I}(\mathcal{B}(y_i \to y_{i+1} \to \cdots) \geq_t \mathrm{val}_{\mathcal{B}}(y_i, M, \mathcal{I}) \geq_t \ell$. Therefore, we can assume that $\mathbf{b}$ is decided at i; hence the evaluation of $\mathbf{b}$ depends on the evaluation of a K-path starting at y. Therefore $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \mathrm{val}_{\mathcal{B}}(y, K, \mathcal{I}) \geq_t \ell$.

Finally, we can assume that y_i is not an internal node of J. If for every j, y_j is not an internal node of J, then $\mathbf{b}$ is a $(K \uparrow M)$-branch. Since $K \uparrow M$ is an extension of K, we have that $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \mathrm{val}_{\mathcal{B}}(y, K, \mathcal{I}) \geq_t \ell$. Therefore, we can assume there is a least j such that y_j is an internal node of J. If $\mathbf{b}$ is transitive at j, then $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \mathrm{val}_{\mathcal{B}}(y_j, M, \mathcal{I}) \geq_t \ell$ by a result we proved above. If $\mathbf{b}$ is decided at j, then $\mathcal{B}(\mathbf{b})$ is decided by a $(K \uparrow M)$-path; hence $\mathcal{I}(\mathcal{B}(\mathbf{b})) \geq_t \mathrm{val}_{\mathcal{B}}(y, K \uparrow M, \mathcal{I}) \geq_t \mathrm{val}_{\mathcal{B}}(y, K, \mathcal{I}) \geq_t \ell$. This concludes that $\mathrm{val}_{\mathcal{B}}(y, M, \mathcal{I}) \geq_t \ell$. Since M is taken arbitrary, this proves that $\mathrm{val}_{\mathcal{B}}(y, J \uparrow K, \mathcal{I}) \geq_t \ell$. □

Lemma 10. *Let $\mathcal{I}$ be an interpretation and $\ell \in \{\mathbf{f}, \mathbf{u}, \mathbf{t}\}$. There is a justification J (not necessarily locally complete) such that*

- $\mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) \geq_t \ell$ *if and only if x is an internal node of J;*
- $\mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}) \geq_t \ell$ *for all internal nodes x of J.*

Proof. Let $X = \{x \in \mathcal{F}_d \mid \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) \geq_t \ell\}$. If $X = \emptyset$, take the empty justification. By Lemma 8, for all $x \in X$, there is a justification J_x with x as internal node such that $\mathrm{val}_{\mathcal{B}}(y, J_x, \mathcal{I}) \geq_t \ell$ for all internal nodes y of J_x. By the well-ordering theorem, fix a well-order on $X = \{x_i \mid i \leq \beta\}$. Define

1. $K_0 = J_{x_0}$;
2. $K_{i+1} = K_i \uparrow J_{x_{i+1}}$ for any ordinal $i < \beta$;
3. $K_\alpha = \bigcup_{i<\alpha} K_i$ for any limit ordinal $\alpha \leq \beta$.

This is an increasing sequence of justifications. We prove for all $i \leq \beta$ that $\mathrm{val}_{\mathcal{B}}(y, K_i, \mathcal{I}) \geq_t \ell$ for all internal nodes y of K_i. We do this by transfinite induction. The base case is trivial by Lemma 8. The successor case is immediate from Lemma 9. Take a limit ordinal $\alpha \leq \beta$, a locally complete extension M of K_α, and an internal node y of K_α. There is an $i < \alpha$ so that y is an internal node of K_i. Since $\mathrm{val}_{\mathcal{B}}(y, K_i, \mathcal{I}) \geq_t \ell$ and M is also an extension of K_i, we have that $\mathrm{val}_{\mathcal{B}}(y, M, \mathcal{I}) \geq_t \ell$. Since M is taken arbitrary, we have that $\mathrm{val}_{\mathcal{B}}(y, K_\alpha, \mathcal{I}) \geq_t \ell$.

By construction K_β contains exactly the elements in X as internal nodes, completing the proof, by setting $J = K_\beta$. □

Theorem 5. *If $\mathcal{B}$ is splittable, then for every interpretation $\mathcal{I}$ there is a locally complete justification J such that $\mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I})$ for all $x \in \mathcal{F}_d$.*

Proof. By Lemma 10, there are justifications $J_{\mathbf{t}}$, $J_{\mathbf{u}}$, and $J_{\mathbf{f}}$ such that

- J_ℓ is ℓ-domain supporting
- $\mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) \geq_t \ell$ if and only if y is an internal node of J_ℓ.

Define K to be equal to $J_{\mathbf{t}} \uparrow J_{\mathbf{u}}$, which is an extension of $J_{\mathbf{t}}$. This means that for all $x \in \mathcal{F}_d$ with $\mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) = \mathbf{t}$, we have that $\mathrm{val}_{\mathcal{B}}(x, K, \mathcal{I}) \geq_t \mathrm{val}_{\mathcal{B}}(x, J_{\mathbf{t}}, \mathcal{I}) = \mathbf{t}$. Hence, $\mathrm{val}_{\mathcal{B}}(x, K, \mathcal{I}) = \mathbf{t} = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I})$. Take $x \in \mathcal{F}_d$ with $\mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) = \mathbf{u}$. By Lemma 9, we have that $\mathrm{val}_{\mathcal{B}}(x, K, \mathcal{I}) \geq_t \mathbf{u}$ because $J_{\mathbf{t}}$ and $J_{\mathbf{u}}$ are $\mathbf{u}$-domain supporting. By the definition of supported value, we have that $\mathrm{val}_{\mathcal{B}}(x, K, \mathcal{I}) \leq_t \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) = \mathbf{u}$; hence $\mathrm{val}_{\mathcal{B}}(x, K, \mathcal{I}) = \mathbf{u} = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I})$. Define J to be equal to $K \uparrow J_{\mathbf{f}}$. Take $x \in \mathcal{F}_d$ such that $\mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) \in \{\mathbf{u}, \mathbf{t}\}$. Since J is an extension of K, we have that $\mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}) \geq_t \mathrm{val}_{\mathcal{B}}(x, K, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I})$; hence $\mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I})$ by definition of supported value. Take $x \in \mathcal{F}_d$ with $\mathrm{SV}_{\mathcal{B}}(x, \mathcal{I}) = \mathbf{f}$. Therefore $\mathrm{val}_{\mathcal{B}}(x, J, \mathcal{I}) = \mathbf{f} = \mathrm{SV}_{\mathcal{B}}(x, \mathcal{I})$.

It is straightforward that J is locally complete since it contains all $x \in \mathcal{F}_d$ as internal nodes. □

Proposition 13. *$\mathcal{B}_{\mathrm{sp}}$, $\mathcal{B}_{\mathrm{KK}}$, $\mathcal{B}_{\mathrm{st}}$, and $\mathcal{B}_{\mathrm{wf}}$ are splittable.*

Proof. $\mathcal{B}_{\mathrm{sp}}$ is totally decided on all branches, and thus is splittable.

$\mathcal{B}_{\mathrm{KK}}$ is splittable because it is transitive.

$\mathcal{B}_{\mathrm{st}}$ is totally transitive on positive branches and negative branches. Take a branch $\mathbf{b}\colon x_0 \to x_1 \to \cdots$ with a first sign switch at i. It is straighforward that $\mathbf{b}$ is decided at $j \geq i$ and transitive at $j < i$. This proves that $\mathbf{b}$ is totally splittable; hence $\mathcal{B}_{\mathrm{st}}$ is splittable.

$\mathcal{B}_{\mathrm{wf}}$ is splittable because it is transitive. □

Proof of Theorem 1. Follows directly from Proposition 13 and Theorem 5. □

It is now left to prove Theorem 2. We first prove this for $\mathcal{B}'_{\mathrm{sp}}$.

Proposition 14. *Let $\mathcal{I}$ be a $\mathcal{B}'_{\mathrm{sp}}$-model. For $x \in \mathcal{F}_d$: $\mathcal{I}(x) = \mathbf{t}$ if and only if there exists a rule $x \leftarrow A$ such that for all $a \in A$: $\mathcal{I}(a) = \mathbf{t}$.*

Proof. If $\mathcal{I}(x) = \mathbf{t}$, we have that $\mathrm{SV}_{\mathcal{B}'_{\mathrm{sp}}}(x, \mathcal{I}) = \mathbf{t}$; hence there is a justification J such that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(x, J, \mathcal{I}) = \mathbf{t}$. Let A be the body of the rule in J for x, i.e. A is the set of direct children of x in J. Take $y \in A$. If $x \to y$ is part of an infinite branch in $B_J(x)$, then it is obvious that $\mathcal{I}(y) = \mathbf{t}$. So assume this is not the case, then we have that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(y, J, \mathcal{I}) = \mathbf{t}$ since every branch in $B_J(y)$ is part of a finite branch in $B_J(x)$ that maps to $\mathbf{t}$ under $\mathcal{I}$. Therefore, $\mathrm{SV}_{\mathcal{B}'_{\mathrm{sp}}}(y, \mathcal{I}) = \mathbf{t}$. Since $\mathcal{I}$ is a $\mathcal{B}'_{\mathrm{sp}}$-model, we get that $\mathcal{I}(y) = \mathbf{t}$.

Assume now that there exists such a rule $x \leftarrow A$. Since $\mathcal{I}(y) = \mathbf{t}$ for all $y \in A$. We have by the above proof that we have a true rule for y. By iteratively adding these rules we get a locally complete justification J. We prove that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(x, J, \mathcal{I}) = \mathbf{t}$. Since every element of this justification is mapped to $\mathbf{t}$ under $\mathcal{I}$, this is straightforward. Now since $\mathcal{I}$ is a $\mathcal{B}'_{\mathrm{sp}}$-model, we get that $\mathcal{I}(x) = \mathbf{t}$. □

Proposition 15. *Let $\mathcal{I}$ be a $\mathcal{B}'_{\mathrm{sp}}$-model. $\mathcal{I}(x) = \mathbf{f}$ if and only if for every rule $x \leftarrow A$ there is an $a \in A$ such that $\mathcal{I}(a) = \mathbf{f}$.*

Proof. The right-hand side is equivalent to $\mathrm{SV}_{\mathcal{B}_{\mathrm{sp}}}(x, \mathcal{I}) = \mathbf{f}$. By consistency of $\mathcal{B}_{\mathrm{sp}}$, this is equivalent to $\mathrm{SV}_{\mathcal{B}_{\mathrm{sp}}}(\sim x, \mathcal{I}) = \mathbf{t}$. By Proposition 14, this is equivalent to $\mathcal{I}(\sim x) = \mathbf{t}$; hence $\mathcal{I}(x) = \mathbf{f}$, which completes the proof. □

Corollary 2. *Every $\mathcal{B}'_{\mathrm{sp}}$-model is a $\mathcal{B}_{\mathrm{sp}}$-model.*

Proof. Follows directly from Propositions 14 and 15. □

Proof of Theorem 2 for $\mathcal{B}'_{\mathrm{sp}}$. Every x with $\mathrm{SV}_{\mathcal{B}'_{\mathrm{sp}}}(x, \mathcal{I}) = \mathbf{t}$ have a true rule $x \leftarrow A$. By combining all these rules, we get a locally complete justification $J_{\mathbf{t}}$ such that for every internal node y of $J_{\mathbf{t}}$ we get that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(y, J_{\mathbf{t}}, \mathcal{I}) = \mathbf{t}$. By a similar construction we get a justification $J_{\mathbf{u}}$. This is not a locally complete justification. However, the justification $J_{\mathbf{t}} \uparrow J_{\mathbf{u}}$ is locally complete. For every internal node y of $J_{\mathbf{t}}$ we have that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(y, J_{\mathbf{t}} \uparrow J_{\mathbf{u}}, \mathcal{I}) = \mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(y, J_{\mathbf{t}}, \mathcal{I}) = \mathbf{t}$. For every internal node y of $J_{\mathbf{u}}$ we have that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(y, J_{\mathbf{t}} \uparrow J_{\mathbf{u}}, \mathcal{I}) = \mathbf{u}$. By extending the justification $J_{\mathbf{t}} \uparrow J_{\mathbf{u}}$ to a complete justification, we do not change the value of the internal nodes of $J_{\mathbf{t}} \uparrow J_{\mathbf{u}}$; hence this provides a justification J such that $\mathrm{val}_{\mathcal{B}'_{\mathrm{sp}}}(y, J, \mathcal{I}) = \mathrm{SV}_{\mathcal{B}'_{\mathrm{sp}}}(y, \mathcal{I})$ for all $y \in \mathcal{F}_d$. □

Proof of Theorem 2 for $\mathcal{B}'_{\mathrm{st}}$. Take $\ell \in \mathcal{L}$. Take $x \in \mathcal{F}_d$ such that $\mathrm{SV}_{\mathcal{B}'_{\mathrm{st}}}(x, \mathcal{I}) = \ell$; hence there is a locally complete justification J_x such that $\mathrm{val}_{\mathcal{B}'_{\mathrm{st}}}(x, J_x, \mathcal{I}) = \ell$. Let J_x^* be the justification J restricted to the domain of elements y reachable from x through a J-path of the same sign (this also means that x and y have the same sign). Every internal node y of J_x^* has $\mathrm{val}_(y, J_x, \mathcal{I}) \geq_t \ell$. The evaluation of every branch in $B_{J_x}(y)$ is equal to the evaluation of a path from x to y with the same sign concatenated with this branch; hence this branch is evaluated to $\geq_t \mathrm{val}_{\mathcal{B}'_{\mathrm{st}}}(x, J_x, \mathcal{I}) \geq_t \ell$; hence $\mathrm{val}_{\mathcal{B}'_{\mathrm{st}}}(y, J_x, \mathcal{I}) \geq_t \ell$. Therefore, $\mathcal{I}(y) \geq_t \ell$. For every leaf y of J_x^*, we have $\mathcal{I}(y) \geq_t \ell$ as well. Indeed, if y is part of an infinite J_x-path starting with x, then this is obvious. If y is not, then $B_{J_x}(y)$ only contains finite branches, and the result follows.

If we paste together all these J_x^* for all $x \in \mathcal{F}_d$ such that $\mathrm{SV}_{\mathcal{B}'_{\mathrm{st}}}(x, \mathcal{I}) = \ell$ provides a not necessarily locally complete justification J_ℓ such that every node y in J_ℓ has $\mathcal{I}(y) \geq_t \ell$.

Moreover, $J_{\mathbf{t}}$ is locally complete. Our final justification is then $J_{\mathbf{t}} \uparrow J_{\mathbf{u}} \uparrow J_{\mathbf{f}}$. By construction everything works out. □