Using SAT and Partial MaxSAT for Reasoning with System Z and System W

Christoph Beierle[1], Aron Spang[1] and Jonas Haldimann[1,2,3]

[1]Knowledge Based Systems, FernUniversität in Hagen, 58084 Hagen, Germany
[2]Institute of Logic and Computation, TU Wien, 1040 Vienna, Austria
[3]University of Cape Town and CAIR, Cape Town, South Africa

Abstract
Nonmonotonic reasoning from conditional belief bases typically depends on a structure over possible worlds that relies on the verification and falsification of conditionals. A major challenge in implementing such reasoning approaches is that the number of worlds in these structures grows exponentially with the number of propositional variables occurring in the belief base. For addressing this problem by using the power of current solvers, recently an implementation of reasoning with system W using Partial MaxSAT problems has been proposed. In this paper, we investigate this approach in more detail, present a formal correctness proof of the system W inference algorithm SWinf, and extend its empirical evaluation. Furthermore, we show that the approach can be transferred to implementing Pearl's system Z by using SAT problems, and prove the correctness of the resulting system Z inference algorithm SZinf. Our implementations of system Z and system W demonstrate that they outperform previous implementations and allow for signature and knowledge base sizes that have been infeasible before.

1. Introduction

Conditionals play a major role in knowledge representation and reasoning, and many different semantics have been proposed for conditional belief bases, like probability distributions, plausibility orderings, possibility distributions, ranking functions and special instances of them, or conditional objects (see, e.g., [1, 2, 3, 4, 5, 6, 7, 8]). However, software systems implementing any of these inference methods have attracted much less attention. Any such implementation of inference with respect to a conditional belief base Δ has to cope with the number of worlds growing exponentially with the number of propositional variables occurring in Δ. For addressing this practical side of nonmonotonic reasoning, we consider Pearl's well-known system Z [9], and system W [10] that exhibits some notable properties like extending system Z and thus rational closure [11], avoiding the drowning problem [12], and fully complying with syntax splitting [13, 14] and also with conditional syntax splitting [15]. Because the first implementation of system W [16] severely limits the number of propositional variables in Δ to about 20 for practical applications, corresponding to about one million worlds, recently an implementation of reasoning with system W using partial MaxSAT problems has been developed, with first evaluation results of up to 60 variables and thus 2^{60} worlds [17].

This paper extends our work presented in [17] in several directions, providing the following main contributions:

- We elaborate the approach in [17] in more detail and provide a formal correctness proof for its system W inference algorithm SWinf.
- We extend its empirical evaluation, demonstrating that it scales up system W inference up to 120 variables and thus 2^{120} worlds and to belief bases of up to 200 conditionals.
- We show how the approach can be transferred, yielding a SAT-based realization of system Z.
- We give a correctness proof for our system Z inference algorithm SZinf.
- We implement system Z correspondingly, outperforming previous implementations and allowing for signature and knowledge base sizes that have not been possible before.

This paper is organized as follows. After briefly recalling the background on conditional logic in Section 2 and system W in Section 3, we present and illustrate our algorithm SWinf for system W in Section 4 and prove its correctness in Section 5. We adapt this approach for system Z in Section 6, illustrate the resulting algorithm SZinf and prove this algorithm's correctness in Section 7. Finally we evaluate the runtimes of our new algorithms for system W and system Z in Section 8 before concluding and pointing out future work in Section 9.

2. Background: Conditional Logic

A *(propositional) signature* is a finite set Σ of propositional variables, and $\mathcal{L}_\Sigma$ denotes the propositional language over Σ. We may denote a conjunction $A \wedge B$ by AB and a negation $\neg A$ by $\overline{A}$. The set of interpretations over a signature Σ, also called worlds, is Ω_Σ. We may identify a world with the corresponding complete conjunction of all elements Σ in either positive or negated form. An $\omega \in \Omega_\Sigma$ is a *model* of $A \in \mathcal{L}_\Sigma$ if A holds in ω, denoted as $\omega \models A$, and the set of models of A is $\mathit{Mod}_\Sigma(A) = \{\omega \in \Omega_\Sigma \mid \omega \models A\}$, sometimes denoted as Ω_A. A formula A *entails* a formula B, written $A \models B$, if $\mathit{Mod}_\Sigma(A) \subseteq \mathit{Mod}_\Sigma(B)$. Furthermore, for $F \in \mathcal{L}_\Sigma$ and $M \subseteq \mathcal{L}_\Sigma$, we use the notation $\omega \models M$ iff $\omega \models F_i$ for every $F_i \in M$; $\Omega_M = \{\omega \in \Omega_\Sigma \mid \omega \models M\}$; $\overline{M} = \{\overline{F}_i \mid F_i \in M\}$; and $F \wedge M = F \wedge F_1 \wedge \ldots \wedge F_m$ for $M = \{F_1, \ldots, F_m\}$.

A *conditional* $(B|A)$ connects two formulas A, B and represents the rule "If A then usually B". The conditional language over Σ is $(\mathcal{L}|\mathcal{L})_\Sigma = \{(B|A) \mid A, B \in \mathcal{L}_\Sigma\}$. A *belief base* Δ is a finite set of conditionals. For a world ω, a conditional $(B|A)$ is either *verified* by ω if $\omega \models AB$, *falsified* by ω if $\omega \models A\overline{B}$, or *not applicable* to ω if $\omega \models \overline{A}$ [18]. An example for semantics of conditionals are functions $\kappa : \Omega_\Sigma \to \mathbb{N}$ such that $\kappa(\omega) = 0$ for at least one $\omega \in \Omega_\Sigma$, called *ranking functions* or *ordinal conditional functions*

22nd International Workshop on Nonmonotonic Reasoning, November 2-4, 2024, Hanoi, Vietnam
christoph.beierle@fernuni-hagen.de (C. Beierle); aron.spang@fernuni-hagen.de (A. Spang); jonas@haldimann.de (J. Haldimann)
0000-0002-0736-8516 (C. Beierle); 0000-0002-2618-8721 (J. Haldimann)

(OCF), introduced (in a more general form) by Spohn (1988). They express degrees of plausibility where a lower degree denotes "less surprising". Each κ uniquely extends to a function $\kappa : \mathcal{L}_\Sigma \to \mathbb{N} \cup \{\infty\}$ with $\kappa(A) = \min\{\kappa(\omega) \mid \omega \models A\}$ where $\min \emptyset = \infty$. A ranking function κ *accepts* a conditional $(B|A)$, written $\kappa \models (B|A)$, if $\kappa(AB) < \kappa(A\overline{B})$, and κ accepts Δ, written $\kappa \models \Delta$, if κ accepts all conditionals in Δ, and Δ is *consistent* if there is a ranking function accepting Δ. Every κ induces a nonmonotonic inference relation $\mid\!\sim^{\kappa}$ between formulas in $\mathcal{L}_\Sigma$, given by

$$A \mid\!\sim^{\kappa} B \quad \text{iff} \quad A \equiv \bot \text{ or } \kappa(AB) < \kappa(A\overline{B}). \tag{1}$$

3. System W

An inductive inference operator completes an explicitly given belief base to the inference relation representing all conditional beliefs an agent can derive [13]. One such inference operator is System W [19, 10] which takes into account the *tolerance information* expressed by the Z-partition (also called *ordered partition*) of a belief base Δ, defined in the following.

Definition 1 (Z-partition $OP(\Delta)$ [9]). *Let $\Delta = \{(B_1|A_1), \dots, (B_n|A_n)\}$ be a belief base. A conditional $(B|A)$ is* tolerated *by a set of conditionals Δ if there exists a world $\omega \in \Omega_\Sigma$ that verifies $(B|A)$ and does not falsify any conditional in Δ, i.e., $\omega \models AB$ and $\omega \models \bigwedge_{i=1}^{n}(\overline{A_i} \vee B_i)$. The* Z-partition *$OP(\Delta) = (\Delta^0, \dots, \Delta^k)$ of a belief base Δ is the partition of Δ where each Δ^i is the (with respect to set inclusion) maximal subset of $\bigcup_{j=i}^{k} \Delta^j$ that is tolerated by $\bigcup_{j=i}^{k} \Delta^j$.*

It is well-known that $OP(\Delta)$ exists iff Δ is consistent; moreover, because the Δ^i are chosen inclusion-maximal, the Z-partition is unique [20]. System W combines the Z-partition with the structural information about which conditionals are falsified by a world, yielding the preferred structure on worlds $<^{\mathsf{w}}_{\Delta}$ underlying system W.

Definition 2 (ξ^j, preferred structure $<^{\mathsf{w}}_{\Delta}$ on worlds [19, 10]). *Let $\Delta = \{(B_1|A_1), \dots, (B_n|A_n)\}$ with $OP(\Delta) = (\Delta^0, \dots, \Delta^k)$. For $j = 0, \dots, k$, the function ξ^j maps worlds to the set of falsified conditionals from the set Δ^j:*

$$\xi^j(\omega) := \{r_i \in \Delta^j \mid \omega \models A_i\overline{B_i}\}, \tag{2}$$

The preferred structure on worlds *is the binary relation $<^{\mathsf{w}}_{\Delta}$ defined by, for $\omega, \omega' \in \Omega_\Sigma$:*

$$\begin{aligned} \omega <^{\mathsf{w}}_{\Delta} \omega' \quad \textit{iff} \quad & \textit{there exists } m \in \{0, \dots, k\} \textit{ such that} \\ & \xi^i(\omega) = \xi^i(\omega') \quad \forall i \in \{m+1, \dots, k\}, \textit{ and} \\ & \xi^m(\omega) \subsetneqq \xi^m(\omega'). \end{aligned} \tag{3}$$

Thus, $\omega <^{\mathsf{w}}_{\Delta} \omega'$ if and only if ω falsifies strictly fewer conditionals than ω' in the partition with the biggest index m where the conditionals falsified by ω and ω' differ.

Definition 3 (system W, $\mid\!\sim^{\mathsf{w}}_{\Delta}$[10]). *Let Δ be a consistent belief base and let $A, B \in \mathcal{L}_\Sigma$ be formulas. Then B is a* system W inference from A (in the context of Δ)*, denoted $A \mid\!\sim^{\mathsf{w}}_{\Delta} B$, if we have:*

$$\begin{aligned} & \textit{for every } \omega' \in \Omega_{A\overline{B}} \\ & \quad \textit{there is an } \omega \in \Omega_{AB} \textit{ such that } \omega <^{\mathsf{w}}_{\Delta} \omega'. \end{aligned} \tag{4}$$

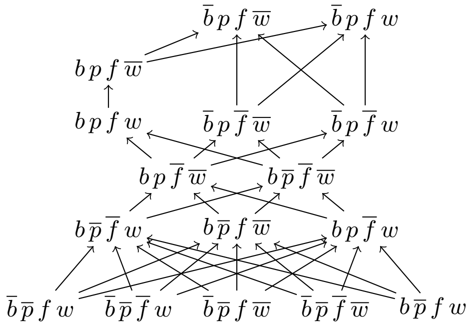

Figure 1: The preferred structure on worlds $<^{\mathsf{w}}_{\Delta_{bird}}$ in Example 1. An edge $\omega \to \omega'$ indicates that $\omega <^{\mathsf{w}}_{\Delta_{bird}} \omega'$; edges that can be obtained from transitivity are omitted.

Example 1 (Δ_{bird}). *Let $\Sigma = \{b, p, f, w\}$ represent birds, penguins, flying things and winged things, and let Δ_{bird} contain $r_1 = (f|b)$, $r_2 = (\overline{f}|p)$, $r_3 = (b|p)$, and $r_4 = (w|b)$. E.g., r_1 expresses* "birds usually fly"*. Then $OP(\Delta_{bird}) = (\Delta^0, \Delta^1)$ with $\Delta^0 = \{(f|b), (w|b)\}$ and $\Delta^1 = \{(\overline{f}|p), (b|p)\}$. Using $<^{\mathsf{w}}_{\Delta_{bird}}$ (Figure 1), we can check that $p \mid\!\sim^{\mathsf{w}}_{\Delta_{bird}} w$ holds, i.e., that penguins usually have wings is a system W inference in the context of Δ_{bird}.*

System W captures system Z in the sense that every entailment that is possible with system Z is also possible with system W, i.e., the system W inferences of a belief base Δ are a superset of the system Z inferences of Δ. Furthermore, there are belief bases where this superset relationship is strict, i.e., where system W licenses strictly more inferences than system Z [10]. System W strictly extends also c-inference [21, 10]; further properties of system W and its relationshsips to other inductive inference operators are described in [22, 23, 24].

For a set M and a partial order $<$ on M, the minimal elements of $N \subseteq M$ are denoted by: $\min(N, <) = \{n \in N \mid$ there is no $n' \in N$ s.t. $n' < n\}$. Because $<^{\mathsf{w}}_{\Delta}$ is a strict partial order [10], Definition 3 directly implies that is suffices to consider only the $<^{\mathsf{w}}_{\Delta}$-minimal worlds for checking whether $\mid\!\sim^{\mathsf{w}}_{\Delta}$ holds.

Proposition 1 ($\mid\!\sim^{\mathsf{w}}_{\Delta}$). *Let Δ be a consistent belief base and $A, B \in \mathcal{L}_\Sigma$. Then*

$$\begin{aligned} A \mid\!\sim^{\mathsf{w}}_{\Delta} B \quad \textit{iff} \quad & \textit{for every } \omega' \in \min(\Omega_{A\overline{B}}, <^{\mathsf{w}}_{\Delta}) \\ & \textit{there is an } \omega \in \min(\Omega_{AB}, <^{\mathsf{w}}_{\Delta}) \\ & \textit{such that } \omega <^{\mathsf{w}}_{\Delta} \omega'. \end{aligned}$$

4. Algorithm SWinf: System W Inference using MaxSAT

In this section we introduce the algorithm SWinf(Δ, A, B) (system W inference with Partial MaxSAT, Algorithm 1) [17] which takes a belief base Δ and two formulas A, B as input and answers the question whether $A \mid\!\sim^{\mathsf{w}}_{\Delta} B$ holds. Implementing system W by computing the relation $<^{\mathsf{w}}_{\Delta}$ [16] will work for small signatures but does not scale well because of the number of worlds to be considered grows exponentially with the size of the signature. Therefore, we will employ Partial MaxSAT concepts [25] in SWinf and utilize the power of current SAT-solvers. Given a set of formulas S of soft constraints and a set of formulas H of hard constraints the *extended partial maximum satisfiability*

Algorithm 1 $\mathsf{SWinf}(\Delta, A, B)$

Input: consistent belief base Δ and formulas A, B
Output: *Yes* if $A \mathrel{|\!\!\sim}^{\mathsf{w}}_{\Delta} B$, and *No* otherwise

```
 1: let OP(Δ) = (Δ^0, ..., Δ^k)
 2: function recWinf(j, H)
 3:   V ← MCS(nf(Δ^j), H ∪ {AB})
 4:   F ← MCS(nf(Δ^j), H ∪ {A B̄})
 5:   if ¬(∀N' ∈ F ∃N ∈ V . N ⊆ N') then
 6:     return No
 7:   for all N ∈ V ∩ F do
 8:     if j = 0 then
 9:       return No
10:     H_new ← (nf(Δ^j) \ N) ∪ N̄
11:     if recWinf(j − 1, H ∪ H_new) = No then
12:       return No
13:   return Yes
14: end function
15: return recWinf(k, ∅)
```

problem $EPMaxSAT(S, H)$ is the optimization problem of maximizing the number of satisfied formulas in S over all interpretations $\omega \in \Omega_H$ and determining all subsets of S that are maximal with this property.

Definition 4 ($MCS(S, H)$)**.** *Let* $S = \{S_1, \ldots, S_s\} \subseteq \mathcal{L}_\Sigma$ *be a set of formulas called* soft constraints*, and let* $H = \{H_1, \ldots, H_h\} \subseteq \mathcal{L}_\Sigma$ *be a set of formulas called* hard constraints*. A* maximal satisfiable subset (MSS) *with respect to* (S, H) *is a set* $M \subseteq S$ *such that* $M \cup H$ *is (classically) consistent and for every* $M' \subseteq S$ *with* $M \subsetneq M'$ *the set* $M' \cup H$ *is not consistent. A set* $N \subseteq S$ *is a* minimal correction subset (MCS) *with respect to* (S, H) *if* $S \setminus N$ *is an MSS w.r.t.* (S, H)*. Then,* $MCS(S, H)$ *denotes the set of all MCS w.r.t.* (S, H)*.*

For using the concepts of MSS and MCS in our context, we rely in particular on the non-falsification of conditionals. For a conditional $(B|A)$, the formula $\overline{A} \vee B$ expressing its *n*on-*f*alsification is denoted by $nf(B|A)$, and nf is extended canonically to a set Δ of condtionals. Thus

$$nf(\Delta) = \{\overline{A} \vee B \mid (B|A) \in \Delta\}$$
$$\overline{nf(\Delta)} = \{A\overline{B} \mid (B|A) \in \Delta\}$$

are the sets of non-falsifying and falsifying formulas, respectively, for the conditionals in Δ.

Example 2. *Let* Δ_{bird} *and* $OP(\Delta_{bird}) = (\Delta^0, \Delta^1)$ *as in Example 1. For* $S = nf(\Delta^1)$ *and* $H = \{pw\}$ *we get* $MSS(nf(\Delta^1), \{pw\}) = MSS(\{\overline{p} \vee \overline{f}, \overline{p} \vee b\}, \{pw\}) = \{\{\overline{p} \vee \overline{f}, \overline{p} \vee b\}\}$*, and thus* $MCS(nf(\Delta^1), \{pw\}) = \{\emptyset\}$*. For* $S = nf(\Delta^0)$ *and* $H = nf(\Delta^1) \cup \{pw\}$ *we get*

$$\begin{aligned} &MSS(nf(\Delta^0), nf(\Delta^1) \cup \{pw\}) \\ =&MSS(\{\overline{b} \vee f, \overline{b} \vee w\}, \{\overline{p} \vee \overline{f}, \overline{p} \vee b, pw\}) \\ =&\{\{\overline{b} \vee w\}\} \end{aligned}$$

and thus $MCS(nf(\Delta^0), nf(\Delta^1) \cup \{pw\}) = \{\{\overline{b} \vee f\}\}$*.*

In addition to using the minima as in Proposition 1 for computing $\mathrel{|\!\!\sim}^{\mathsf{w}}_{\Delta}$, SWinf exploits the fact that the underlying relation $\omega <^{\mathsf{w}}_{\Delta} \omega'$ (cf. Equation (2)) can be determined by iteratively considering the subbases in $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$ starting from the the highest partition element Δ^k. If $\xi^j(\omega) \neq \xi^j(\omega')$ we can decide whether $\omega <^{\mathsf{w}}_{\Delta} \omega'$ or $\omega \not<^{\mathsf{w}}_{\Delta} \omega'$ holds after considering ξ^j; only in case that $\xi^l(\omega) = \xi^l(\omega')$ for all $l \in \{j, \ldots, k\}$ we continue by considering the next lower element Δ^{j-1}.

Assume we have Δ with $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$. The central part of SWinf is the recursive function $recWinf$ that takes the index for one of the sets in the Z-partition and a set of formulas as arguments. It is initially called for the last part Δ^k of the Z-partition. Each set in $\mathcal{V} = MCS(nf(\Delta^k), \{AB\})$ corresponds to a maximal selection of conditionals in Δ^k such that there is a model of AB not falsifying them, and thus to a selection of worlds that falsify a minimal set of conditionals in Δ^k. Analogously, this holds for $\mathcal{F} = MCS(nf(\Delta^k), \{A\overline{B}\})$ and $A\overline{B}$. If there is an $N \in \mathcal{F}$ for which there is no $N' \in \mathcal{V}$ with $N \subseteq N'$, there is a world $\omega' \in Mod_\Sigma(A\overline{B})$ for which there is no world $\omega \in Mod_\Sigma(AB)$ with $\xi^k(\omega) \subseteq \xi^k(\omega')$. Thus, if the condition in Line 5 holds, we have that $A \mathrel{|\!\!\not\sim}^{\mathsf{w}}_{\Delta} B$. Otherwise, we continue to consider the intersection $\mathcal{V} \cap \mathcal{F}$. Each set $N \in \mathcal{V} \cap \mathcal{F}$ corresponds to a selection of conditionals that is a minimal set of falsified conditionals in Δ^k both for some models of AB and $A\overline{B}$. If there is no such N, then all inclusions we considered in Line 5 are strict inclusions and we have $A \mathrel{|\!\!\sim}^{\mathsf{w}}_{\Delta} B$. Otherwise, for each such N we need to consider the parts of the Z-partition with lower indices to check whether for each world $\omega' \in Mod_\Sigma(AB)$ that falsifies the conditionals in N there is a world $\omega \in Mod_\Sigma(A\overline{B})$ that falsifies the conditionals in N. To do this, we add $(nf\Delta^k \setminus N) \cup \overline{N}$ as hard constraints to fix the falsification behaviour on Δ^k and call the function $recWinf$ recursively.

Example 3. *Executing* $\mathsf{SWinf}(\Delta_{bird}, p, w)$ *results in two successive calls of* $recWinf$ *involving the following values and conditions, cf. Example 2:*

$recWinf(j = 1, H = \emptyset)$
$\mathcal{V} = \{\emptyset\}$
$\mathcal{F} = \{\emptyset\}$
$(\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V} \,.\, N \subseteq N') = true$
$\mathcal{V} \cap \mathcal{F} = \{\emptyset\}$
$j = 1 > 0$
return Yes *iff* $recWinf(0, \{\overline{p} \vee \overline{f}, \overline{p} \vee b\}) =$ Yes

$recWinf(j = 0, H = \{\overline{p} \vee \overline{f}, \overline{p} \vee b\})$
$\mathcal{V} = \{\{\overline{b} \vee f\}\}$
$\mathcal{F} = \{\{\overline{b} \vee f, \overline{b} \vee w\}\}$
$(\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V} \,.\, N \subseteq N') = true$
$\mathcal{V} \cap \mathcal{F} = \emptyset \rightarrow$ *return* Yes

Thus, $\mathsf{SWinf}(\Delta_{bird}, p, w)$ *returns* Yes*, and* $p \mathrel{|\!\!\sim}^{\mathsf{w}}_{\Delta_{bird}} w$*.*

When asking whether $\overline{w}$ *can be inferred from* p *in the context of* Δ_{bird} *with system W,* $\mathsf{SWinf}(\Delta_{bird}, p, \overline{w})$ *yields* No*:*

$recWinf(j = 1, H = \emptyset)$
$\mathcal{V} = \{\emptyset\}$
$\mathcal{F} = \{\emptyset\}$
$(\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V} \,.\, N \subseteq N') = true$
$\mathcal{V} \cap \mathcal{F} = \{\emptyset\}$
$j = 1 > 0$
return Yes *iff* $recWinf(0, \{\overline{p} \vee \overline{f}, \overline{p} \vee b\}) =$ Yes

$recWinf(j = 0, H = \{\overline{p} \vee f, \overline{p} \vee b\})$
$\mathcal{V} = \{\{\overline{b} \vee f, \overline{b} \vee w\}\}$
$\mathcal{F} = \{\{\overline{b} \vee f\}\}$
$(\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V} \,.\, N \subseteq N') = false \rightarrow$ *return* No

5. Correctness Proof for SWinf

An important characteristic of the search space of the algorithm SWinf is described by the falsification and non-falsification behaviour of the conditionals in $\Delta^{j+1} \cup \cdots \cup \Delta^k$ for a given $j \in \{0, \ldots, k\}$. For formally characterizing this behaviour we use the following notion.

Definition 5 (*nf/f*-condition for (Δ, j))**.** *Let $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$, and let $j \in \{0, \ldots, k\}$. A set of formulas H is a non-falsifying/falsifying condition (nf/f-condition) for (Δ, j) if there are, for $i \in \{j+1, \ldots, k\}$ sets $\Delta^i_{nf}, \Delta^i_f \subseteq \Delta^i$ such that $\Delta^i = \Delta^i_{nf} \cup \Delta^i_f$ and $\Delta^i_{nf} \cap \Delta^i_f = \emptyset$, and*

$$H = \bigcup_{i \in \{j+1, \ldots, k\}} nf(\Delta^i_{nf}) \cup \overline{nf(\Delta^i_f)}$$

Thus, an *nf/f*-condition H for (Δ, j) contains either the non-falsifying formula $\overline{A} \vee B$ or the falsifying formula $A\overline{B}$ for every conditional $(B|A) \in \Delta^{j+1} \cup \ldots \cup \Delta^k$; this way the *nf/f*-condition H ensures that for any two worlds $\omega, \omega' \in \Omega_\Sigma$ with $\omega \models H$ and $\omega' \models H$ we have $\xi^l(\omega) = \xi^l(\omega')$ for all $l \in \{j+1, \ldots, k\}$.

To prove the correctness of SWinf, we prove the following three lemmas first. Lemma 1 describes the condition in Line 5 of SWinf.

Lemma 1. *Let Δ be a consistent belief base with $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$, let $j \in \{0, \ldots, k\}$, $A, B \in \mathcal{L}_\Sigma$, and H be an nf/f-condition for (Δ, j). Let*

$$\mathcal{V} = MCS(nf(\Delta^j), H \cup \{AB\}) \text{ and}$$
$$\mathcal{F} = MCS(nf(\Delta^j), H \cup \{A\overline{B}\}).$$

Then $\neg(\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V} . N \subseteq N')$ holds iff there is a world $\omega' \in Mod_\Sigma(H \wedge A\overline{B})$ such that for all $\omega \in Mod_\Sigma(H \wedge AB)$ it holds that $\xi^j(\omega) \not\subseteq \xi^j(\omega')$.

Proof. **Direction $\Rightarrow$:** Assume $\neg(\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V} . N \subseteq N')$; this is equivalent to $\exists N' \in \mathcal{F}\, \forall N \in \mathcal{V} . N \not\subseteq N'$. Let $N' \in \mathcal{F}$ be such a set with $\forall N \in \mathcal{V} . N \not\subseteq N'$.

Let $\omega' \in Mod_\Sigma(H \wedge A\overline{B})$ be a world that satisfies $nf(\Delta^i) \setminus N'$; such a world exists because $nf(\Delta^i) \setminus N'$ is consistent by the construction of MCS. Let ω be any world in $Mod_\Sigma(H \wedge AB)$. Let $N \in \mathcal{V}$ such that $N \subseteq nf(\xi^j(\omega))$; such an N exists because $nf(\xi^j(\omega))$ is a (not necessarily minimal) correction set. By assumption, $N \not\subseteq N'$. Therefore, also $nf(\xi^j(\omega)) \not\subseteq N'$. Because MCS yields *minimal* correction sets, ω' falsifies all conditionals $c \in \Delta^j$ with $nf(c) \in N'$, and ω' does not falsify the conditionals $d \in \Delta^j$ with $nf(d) \notin N'$; in summary $nf(\xi^j(\omega)) = N'$. With $nf(\xi^j(\omega)) \not\subseteq N'$ we have $\xi^j(\omega) \not\subseteq \xi^j(\omega')$.

Direction $\Leftarrow$: We prove this direction by contraposition. Assume that $\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V} . N \subseteq N'$. Let ω' be any world in $Mod_\Sigma(H \wedge A\overline{B})$; we need to show that there is an $\omega \in Mod_\Sigma(H \wedge AB)$ such that $\xi^j(\omega) \subseteq \xi^j(\omega')$.

Let $N' \in \mathcal{F}$ such that $N' \subseteq nf(\xi^j(\omega'))$; such an N' exists because $nf(\xi^j(\omega'))$ is a (not necessarily minimal) correction set. By assumption there is an $N \in \mathcal{V}$ such that $N \subseteq N'$. Let $\omega \in Mod_\Sigma(nf(\Delta^j) \setminus N)$; such a world exists because $nf(\Delta^j) \setminus N$ is consistent by the definition of MCS. Because MCS yields *minimal* correction sets, ω falsifies all conditionals $c \in N\Delta^j$ with $nf(c) \in N$. Furthermore, ω' falsifies all conditionals $d \in \Delta^j$ with $nf(d) \in N'$. Therefore, $N \subseteq N'$ implies $\xi^j(\omega) \subseteq \xi^j(\omega')$. ☐

Lemma 2 describes conditions on sets of falsified conditionals that require a certain set S to be in $\mathcal{V} \cap \mathcal{F}$.

Lemma 2. *Let Δ be a consistent belief base with $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$, let $j \in \{0, \ldots, k\}$, $A, B \in \mathcal{L}_\Sigma$, and let H be an nf/f-condition for (Δ, j). Let*

$$\mathcal{V} = MCS(nf(\Delta^j), H \cup \{AB\}) \text{ and}$$
$$\mathcal{F} = MCS(nf(\Delta^j), H \cup \{A\overline{B}\}).$$

Let $\omega \in Mod_\Sigma(H \wedge AB)$ such that there is no $\omega^* \in Mod_\Sigma(H \wedge AB)$ with $\xi^j(\omega^*) \subsetneqq \xi^j(\omega)$, and let $\omega' \in Mod_\Sigma(H \wedge A\overline{B})$ such that there is no $\omega'^* \in Mod_\Sigma(H \wedge AB)$ with $\xi^j(\omega'^*) \subsetneqq \xi^j(\omega')$. Then $\xi^j(\omega) = \xi^j(\omega')$ implies that there is an $S \in \mathcal{V} \cap \mathcal{F}$ with $\omega, \omega' \models nf(\Delta^j) \setminus S$.

Proof. Assume that $\xi^j(\omega) = \xi^j(\omega')$. Then $S = nf(\xi^j(\omega))$ is a correction set with respect to $(nf(\Delta^j), H \cup \{AB\})$, because ω is a model of all $nf(r)$ with $r \in \Delta^j \setminus \xi^j(\omega)$. We need to show that S is a minimal correction set. Towards a contradiction assume that there is an $R \in nf(\xi^j(\omega))$ such that $C = (H \cup \{AB\}) \cup nf(\Delta^j) \setminus nf(\xi^j(\omega)) \cup \{R\}$ is consistent. Let ω^c be a model of C. The world ω^c does not falsify any of the conditionals in $\Delta^j \setminus \xi^j(\omega)$ because it is a model of $nf(\Delta^j) \setminus nf(\xi^j(\omega))$. There must be an $r \in \xi^j(\omega)$ such that $nf(r) = R$. The world ω^c also does not falsify r because it is a model of R. Therefore, $\xi^j(\omega^c) \subsetneqq \xi^j(\omega)$. This contradicts that there is no $\omega^* \in Mod_\Sigma(H \wedge AB)$ with $\xi^j(\omega^*) \subsetneqq \xi^j(\omega)$; therefore S is indeed a minimal correction set with respect to $(nf(\Delta^j), H \cup \{AB\})$.

Analogously we can show that $S = nf(\xi^j(\omega')) = nf(\xi^j(\omega'))$ is a minimal correction set with respect to $(nf(\Delta^j), H \cup \{A\overline{B}\})$. In summary, $S \in \mathcal{V} \cap \mathcal{F}$ and $\omega, \omega' \models nf(\Delta^j) \setminus S$. ☐

Lemma 3 describes an effect of a set $S \in \mathcal{V} \cap \mathcal{F}$ on sets of falsified conditionals.

Lemma 3. *Let Δ be a consistent belief base with $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$, let $j \in \{0, \ldots, k\}$, $A, B \in \mathcal{L}_\Sigma$, and let H be an nf/f-condition for (Δ, j). Let*

$$\mathcal{V} = MCS(nf(\Delta^j), H \cup \{AB\}) \text{ and}$$
$$\mathcal{F} = MCS(nf(\Delta^j), H \cup \{A\overline{B}\}).$$

Then, for $S \in \mathcal{V} \cap \mathcal{F}$ and every $\omega \in Mod_\Sigma(H \wedge AB)$ with $\omega \models nf(\Delta^j) \setminus S$, and every $\omega' \in Mod_\Sigma(H \wedge A\overline{B})$ with $\omega' \models nf(\Delta^j) \setminus S$, it holds that $\xi^j(\omega) = \xi^j(\omega') = \{r \in \Delta^j \mid nf(r) \in S\}$.

Proof. Let $R_S = \{r \in \Delta^j \mid nf(r) \in S\}$. Because $\omega \models nf(\Delta^j) \setminus S$, we have that ω falsifies no conditionals in $\Delta^j \setminus R_S$, i.e., $\xi^j(\omega) \subseteq R_S$. Because S is a *minimal* correction set in $\mathcal{V}$, there is no $r \in R_S$ that does is not falsified by ω: if there were such an r then $S \setminus \{nf(r)\}$ would be an even smaller correction set. Therefore, $\xi^j(\omega) = R_S$.

Analogously we can show that $\xi^j(\omega') = R_S$. ☐

Now we can use Lemma 1, Lemma 2, and Lemma 3 to show Proposition 2 on the output of the recursive algorithm *recWinf*.

Proposition 2. *Let Δ be consistent with $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$, let $j \in \{0, \ldots, k\}$, $A, B \in \mathcal{L}_\Sigma$, and let*

H be an nf/f-condition for (Δ, j). Then $recWinf(j, H)$ returns "Yes" iff

$$\begin{aligned}&\text{for every } \omega' \in Mod_\Sigma(H \wedge A\overline{B})\\&\quad\text{there is an } \omega \in Mod_\Sigma(H \wedge AB) \text{ with } \omega <^{\mathsf{w}}_\Delta \omega'.\end{aligned} \tag{5}$$

Proof. We prove this by induction over j. Let

$$\mathcal{V} = MCS(nf(\Delta^j), H \cup \{AB\}) \text{ and}$$
$$\mathcal{F} = MCS(nf(\Delta^j), H \cup \{A\overline{B}\})$$

as assigned in Lines 3 and 4 (Algorithm 1).

Base Case ($j = 0$): By observing Lines 5 – 9 and 13 we see that $recWinf(j, H)$ returns "Yes" iff

$$\text{not } \neg(\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V}\,.\, N \subseteq N') \text{ and } \mathcal{V} \cap \mathcal{F} = \emptyset. \tag{6}$$

We show that (6) is equivalent to (5) by showing both directions of this equivalence.

Direction $\Rightarrow$: Assume (6) holds. Let ω' be any world in $Mod_\Sigma(H \wedge A\overline{B})$. Because H is an *nf/f*-condition for $(\Delta, 0)$, every model of H falsifies the same conditionals in $\Delta^1, \ldots, \Delta^k$. Therefore, we need to show that there is a world $\omega \in Mod_\Sigma(H \wedge AB)$ with $\xi^0(\omega) \subsetneqq \xi^0(\omega')$.

W.l.o.g. assume that there is no ω'^* with $\xi^0(\omega'^*) \subsetneqq \xi^0(\omega')$. With Lemma 1 and the first part of (6) we have that there is an $\omega \in Mod_\Sigma(H \wedge AB)$ with $\xi^0(\omega) \subseteq \xi^0(\omega')$. W.l.o.g. assume that there is no ω^* with $\xi^0(\omega^*) \subsetneqq \xi^0(\omega)$. Towards a contradiction, assume that $\xi^0(\omega) = \xi^0(\omega')$. With Lemma 2 we have that there is an $S \in \mathcal{V} \cap \mathcal{F}$ which contradicts (6). Therefore, $\xi^0(\omega) \subsetneqq \xi^0(\omega')$.

Direction $\Leftarrow$: Assume that (5) holds. Because H is an *nf/f*-condition for $(\Delta, 0)$, every model of H falsifies the same conditionals in $\Delta^1, \ldots, \Delta^k$. Therefore, with Lemma 1 it follows that the part left of the *and* in (6) holds.

It is left to show $\mathcal{V} \cap \mathcal{F} = \emptyset$. Towards a contradiction assume that $\mathcal{V} \cap \mathcal{F} \neq \emptyset$, i.e., there is an $S \in \mathcal{V} \cap \mathcal{F}$. Because S is a correction set of $\mathcal{F}$, there is an $\omega' \in Mod_\Sigma(H \wedge A\overline{B})$ with $\omega' \models nf(\Delta^0) \setminus S$. By (5) there is an $\omega \in Mod_\Sigma(H \wedge AB)$ with $\omega <^{\mathsf{w}}_\Delta \omega'$. Because $\omega <^{\mathsf{w}}_\Delta \omega'$ and H is an *nf/f*-condition for $(\Delta, 0)$ we have $\xi^0(\omega) \subseteq \xi^0(\omega')$ and therefore $\omega \models nf(\Delta^0) \setminus S$. W.l.o.g. assume that there is no $\omega^* \in Mod_\Sigma(H \wedge AB)$ with $\omega^* \models nf(\Delta^0) \setminus S$ and $\xi^0(\omega^*) \subsetneqq \xi^0(\omega)$. By Lemma 3 we have $\xi^0(\omega) = \xi^0(\omega')$. This implies $\xi(\omega) = \xi(\omega')$ and contradicts $\omega <^{\mathsf{w}}_\Delta \omega'$. Therefore, $\mathcal{V} \cap \mathcal{F} = \emptyset$ and (6) holds.

Induction Step: Let $j > 0$ and assume the proposition holds for $j' = j - 1$. We show that $recWinf(j, H)$ returns "Yes" iff (5) holds by showing both directions of this equivalence.

Direction $\Rightarrow$: Assume that $recWinf(j, H)$ returns "Yes"; therefore the algorithm reaches Line 13 at some point. It is left to show that (5) holds. Let ω' be any world in $Mod_\Sigma(H \wedge A\overline{B})$. W.l.o.g. assume that there is no ω'^* with $\xi^j(\omega'^*) \subsetneqq \xi^j(\omega')$.

The algorithm passes Lines 5 and 6 without returning "No". Therefore, there is a world $\omega \in Mod_\Sigma(H \wedge AB)$ such that $\xi^j(\omega) \subseteq \xi^j(\omega')$. W.l.o.g. assume that there is no ω^* with $\xi^j(\omega^*) \subsetneq \xi^j(\omega)$. We can distinguish two cases.

Case 1: $\xi^j(\omega) \subsetneqq \xi^j(\omega')$.
Because H is an *nf/f*-condition for (Δ, j), all models of H, including ω and ω', falsify the same conditionals in $\Delta^{j+1}, \ldots, \Delta^k$. Therefore, $\omega <^{\mathsf{w}}_\Delta \omega'$.

Case 2: $\xi^j(\omega) = \xi^j(\omega')$.
By Lemma 2 there is an $S \in \mathcal{V} \cap \mathcal{F}$ with $\omega, \omega' \models nf(\Delta^j) \setminus S$. Because S is a minimal correction set in both $\mathcal{V}$ and $\mathcal{F}$, we have $\omega, \omega' \models \overline{S}$. Because the algorithm passes Lines 7–12 without returning "No", for $H_{new} \leftarrow (nf(\Delta^j) \setminus S) \cup \overline{S}$ the function call $recWinf(j - 1, H \cup H_{new})$ returns "Yes". By construction, $H \cup H_{new}$ is an *nf/f*-condition for $(\Delta, j - 1)$. Furthermore, $\omega \models H \wedge AB$. Using the induction hypothesis, there is an $\omega^a \in Mod_\Sigma(H \wedge AB)$ with $\omega^a <^{\mathsf{w}}_\Delta \omega'$.

In summary, (5) holds.

Direction $\Leftarrow$: Assume that (5) holds. We need to show that $recWinf(j, H)$ returns "Yes". With Lemma 1 and because H is an *nf/f*-condition for (Δ, j), we have that $\neg(\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V}\,.\, N \subseteq N')$ does not hold; therefore the algorithm passes Lines 5 and 6 without returning "No".

It is left to show that the algorithm passes Lines 7–12 without returning "No". Let S be any set in $\mathcal{V} \cap \mathcal{F}$. By construction, H_{new} is an *nf/f*-condition for $(\Delta, j - 1)$. Let ω' be any world in $Mod_\Sigma(H \wedge H_{new} \wedge A\overline{B})$. Because $\omega' \models H_{new}$ we have $\omega \models nf(\Delta^j) \setminus S$. Because of (5), there is a world $\omega \in Mod_\Sigma(H \wedge AB)$ with $\omega <^{\mathsf{w}}_\Delta \omega'$. Using $\forall N' \in \mathcal{F}\, \exists N \in \mathcal{V}\,.\, N \subseteq N'$ again, we know that $\xi(\omega) \subseteq \xi(\omega') = \{r \in \Delta^j \mid nf(r) \in S\}$ (the last equation holding due to Lemma 3); therefore $\omega \models nf(\Delta^j) \setminus S$. Using Lemma 3 again, we have $\xi^j(\omega) = \{r \in \Delta^j \mid nf(r) \in S\}$ and therefore $\omega \models H_{new}$. Hence, for every $\omega' \in Mod_\Sigma(H \wedge H_{new} \wedge A\overline{B})$ there is an $\omega \in Mod_\Sigma(H \wedge H_{new} \wedge AB)$ with $\omega <^{\mathsf{w}}_\Delta \omega'$. Employing the induction hypothesis, we have that $recWinf(j - 1, H \cup H_{new})$ returns "Yes".

Thus, algorithm passes Lines 7–12 without returning "No", and then returns "Yes" in Line 13. ☐

Now it is straightforward to show the correctness of SWinfwith Proposition 2.

Theorem 1. *Given a consistent belief base Δ and $A, B \in \mathcal{L}_\Sigma$, the call* SWinf$(\Delta, A, B)$ *always terminates, and it returns "Yes" iff $A \mid\!\sim^{\mathsf{w}}_\Delta B$.*

Proof. SWinf(Δ, A, B) terminates because $\mathcal{V}, \mathcal{F}$ and thus also $\mathcal{V} \cap \mathcal{F}$ are always finite sets, and for every recursive call the index j is decreased by one. SWinf(Δ, A, B) returns "Yes" iff $recWinf(k, \emptyset)$ returns "Yes". Because $\emptyset$ is an *nf/f*-condition for (Δ, k), according to Proposition 2, this happens iff for every $\omega \in Mod_\Sigma(\emptyset \wedge A\overline{B})$ there is an $\omega' \in Mod_\Sigma(\emptyset \wedge AB)$ with $\omega <^{\mathsf{w}}_\Delta \omega'$, which is equivalent to $A \mid\!\sim^{\mathsf{w}}_\Delta B$. ☐

6. Algorithm SZinf for System Z

Pearl's system Z [20] is a well-known inductive inference operator that was shown to coincide with rational closure [26, 27]

Definition 6 (System Z [20])**.** *Let Δ be consistent with $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$. The ranking function κ^z_Δ is defined as follows: If $\omega \in \Omega_\Sigma$ does not falsify any conditional in Δ, then let $\kappa^z_\Delta(\omega) := 0$. Otherwise, let Δ^j be the last part in $OP(\Delta)$ that contains a conditional falsified by ω, and let $\kappa^z_\Delta(\omega) := j + 1$. System Z maps Δ to the inference relation $\mid\!\sim^z_\Delta$ induced by κ^z_Δ according to (1).*

For any consistent Δ, the function κ^z_Δ is the unique least ranking model of Δ [20].

Example 4. *For the belief base Δ_{bird} from Example 1 we have $OP(\Delta_{bird}) = ({\Delta_{bird}}^0, {\Delta_{bird}}^1)$ with ${\Delta_{bird}}^0 = \{r_1, r_4\}$ and ${\Delta_{bird}}^1 = \{r_2, r_3\}$. The*

ω	$\kappa^z_{\Delta_{bird}}(\omega)$
$b\,p\,f\,w$	2
$b\,p\,f\,\overline{w}$	2
$b\,p\,\overline{f}\,w$	1
$b\,p\,\overline{f}\,\overline{w}$	1
$b\,\overline{p}\,f\,w$	0
$b\,\overline{p}\,f\,\overline{w}$	1
$b\,\overline{p}\,\overline{f}\,w$	1
$b\,\overline{p}\,\overline{f}\,\overline{w}$	1
$\overline{b}\,p\,f\,w$	2
$\overline{b}\,p\,f\,\overline{w}$	2
$\overline{b}\,p\,\overline{f}\,w$	2
$\overline{b}\,p\,\overline{f}\,\overline{w}$	2
$\overline{b}\,\overline{p}\,f\,w$	0
$\overline{b}\,\overline{p}\,f\,\overline{w}$	0
$\overline{b}\,\overline{p}\,\overline{f}\,w$	0
$\overline{b}\,\overline{p}\,\overline{f}\,\overline{w}$	0

Table 1
System Z ranking function $\kappa^z_{\Delta_{bird}}$ for Δ_{bird} in Example 4.

Algorithm 2 $\mathsf{SZinf}(\Delta, A, B)$

Input: consistent belief base Δ and formulas A, B
Output: *Yes* if $A \mathrel{|\!\sim}^z_\Delta B$, and *No* otherwise

```
1:  let OP(Δ) = (Δ^0, ..., Δ^k)
2:  function recZinf(j)
3:    V ← SAT(⋃_{i=j}^{k} nf(Δ^i) ∪ {AB})
4:    if V = UNSAT then
5:      return No
6:    F ← SAT(⋃_{i=j}^{k} nf(Δ^i) ∪ {A B̄})
7:    if F = SAT then
8:      if j = 0 then
9:        return No
10:     if recZinf(j − 1) = No then
11:       return No
12:   return Yes
13: end function
14: return if A ≡ ⊥ then Yes else recZinf(k)
```

Z-ranking function induced by Δ_{bird} is displayed in Table 1. We have $\kappa^z_{\Delta_{bird}}(\overline{p}\overline{b}) = 0$ and $\kappa^z_{\Delta_{bird}}(p\overline{b}) = 2$, and thus $\overline{b} \mathrel{|\!\sim}^z_{\Delta_{bird}} \overline{p}$. Analogously, we can check that $\kappa^z_{\Delta_{bird}}(wp) = \kappa^z_{\Delta_{bird}}(\overline{w}p) = 1$, yielding $p \not\mathrel{|\!\sim}^z_{\Delta_{bird}} w$.

Note that this also illustrate that system Z, in contrast to, e.g., system W, suffers from the drowning problem [12] because the birds property of having wings is drowned for penguins because penguins are exceptional birds with respect to the property flying.

Implementations of system Z have to cope with the exponentially growing number of worlds for larger signatures. Based on the approach of SWinf we design a similar algorithm SZinf (Algorithm 2) for system Z.

One of the main differences between system W and system Z is that, while $<^{\mathsf{w}}_\Delta$ compares the sets of conditionals falsified by each world, κ^z_Δ only takes into account the latest j for which any conditional in Δ^j is falsified by each world. This simplifies the algorithm SZinf compared to SWinf in two ways.

First, instead of finding a minimal set of falsified conditionals in Δ^j with $MCS(nf(\Delta^j), H \cup \{A\dot{B}\})$, it is sufficient to check if there is a world ω with $\omega \models A\dot{B}$ that falsifies no conditionals in Δ^j for each $\dot{B} \in \{B, \overline{B}\}$. This does not even require solving a Partial MaxSAT problem; instead we just have to check two sets of formulas for their satisfiability.

Second, because there are no different minimal sets of falsified conditionals to choose from, we do not need an argument H of the recursive function to keep track of the (non-)falsified conditionals in the previously considered parts of $OP(\Delta)$. Instead, we straightforwardly check the satisfiability of $\bigcup_{i=j}^{k} nf(\Delta^i) \cup \{A\dot{B}\}$.

For determining whether A entails B, $\mathsf{SZinf}(\Delta, A, B)$ considers the parts in $OP(\Delta)$ beginning with the latest Δ^k. The main part of SZinf is a recursive function $recZinf(j)$ whose argument is the index of the part of $OP(\Delta)$ that is considered next. For each Δ^j, the algorithm determines $V^j = SAT(\bigcup_{i=j}^{k} nf(\Delta^i) \cup \{AB\})$ and $F^j = SAT(\bigcup_{i=j}^{k} nf(\Delta^i) \cup \{A\overline{B}\})$. Note that for $j < k$, $recZinf(j)$ is only called if $F^{j+1} = SAT$ (cf. Line 7), implying that $\kappa^z_\Delta(A\overline{B}) < j + 2$. If $V = UNSAT$, then $\kappa^z_\Delta(AB) \geqslant j + 1$ and thus $\kappa^z_\Delta(AB) \not< \kappa^z_\Delta(A\overline{B})$, implying $A \not\mathrel{|\!\sim}^z_\Delta B$. If $V = SAT$ and $F = UNSAT$, then we can conclude that $\kappa^z_\Delta(AB) < j + 1$ and $\kappa^z_\Delta(A\overline{B}) \geqslant j + 1$ and thus $\kappa^z_\Delta(AB) < \kappa^z_\Delta(A\overline{B})$, implying $A \mathrel{|\!\sim}^z_\Delta B$. If $V = SAT$ and $F = SAT$, we continue by recursively calling $recZinf(j-1)$ to check the parts of $OP(\Delta)$ with a lower index. Line 14 contains the initial call to $recZinf$ and the handling of the border case that the antecedent is an inconsistent formula.

Note that the function $recZinf$ in SZinf can be easily rewritten to use a loop instead of recursion, but we decided to present it with recursive calls to point out the similarity to SWinf.

Example 5. *Executing $\mathsf{SZinf}(\Delta_{bird}, p, w)$ yields two successive calls of $recZinf$ involving as values and conditions:*

$recZinf(j = 1)$,
$V \leftarrow SAT$,
$V \neq UNSAT$,
$F \leftarrow SAT$,
$F = SAT$,
$j \neq 0$,
return No *iff* $recZinf(0) =$ No

$recZinf(j = 0)$,
$V \leftarrow UNSAT$,
$V = UNSAT \quad \Rightarrow$ *return* No

Thus, $\mathsf{SZinf}(\Delta_{bird}, p, w)$ returns No, *and $p \not\mathrel{|\!\sim}^z_{\Delta_{bird}} w$.*

7. Correctness Proof for SZinf

For proving the correctness of SZinf, we first show the following lemma connecting the satisfiablility of

$$S^j_A = \bigcup_{i=j}^{k} nf(\Delta^i) \cup \{A\}$$

to the rank of a formula A.

Lemma 4. *Let Δ be a consistent belief base with $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$, let $j \in \{0, \ldots, k\}$, and let $A \in \mathcal{L}_\Sigma$.*

1. *If S^j_A is consistent, then $\kappa^z_\Delta(S) < j + 1$.*
2. *If S^j_A is not consistent, then $\kappa^z_\Delta(S) \geqslant j + 1$.*

Proof. **Ad (1.):** Assume that S^j_A is consistent. Then there is a world ω with $\omega \models \bigcup_{i=j}^k nf(\Delta^i)$ and $\omega \models A$. Because $\omega \models \bigcup_{i=j}^k nf(\Delta^i)$, the world ω does not falsify any conditional in $\bigcup_{i=j}^k \Delta^i$, and thus $\kappa^z_\Delta(\omega) < j + 1$. Because $\omega \models A$ and $\kappa^z_\Delta(A) = \min\{\kappa^z_\Delta(\omega) \mid \omega \models A\}$, this implies $\kappa^z_\Delta(A) < j + 1$.

Ad (2.): Assume that S^j_A is not consistent. Then every world ω with $\omega \models A$ falsifies at least one conditional in $\bigcup_{i=j}^k \Delta^i$. Therefore, $\kappa^z_\Delta(A) \geqslant j + 1$. □

Now we can prove the correctness of *recZinf*.

Proposition 3. *Let Δ be a consistent belief base with $OP(\Delta) = (\Delta^0, \ldots, \Delta^k)$, let $j \in \{0, \ldots, k\}$, and let $A, B \in \mathcal{L}_\Sigma$. If S^{j+1}_{AB} and $S^{j+1}_{A\overline{B}}$ are consistent, then recZinf(j) returns "Yes" iff $\kappa^z_\Delta(AB) < \kappa^z_\Delta(A\overline{B})$.*

Proof. We prove this by induction over j.

Base Case ($j = 0$): By assumption S^{j+1}_{AB} and $S^{j+1}_{A\overline{B}}$ are consistent, and thus, by Lemma 4 we have $\kappa^z_\Delta(AB) < j + 2 = 2$ and $\kappa^z_\Delta(A\overline{B}) < j + 2 = 2$, implying that $\kappa^z_\Delta(AB), \kappa^z_\Delta(A\overline{B}) \in \{0, 1\}$.

If S^j_{AB} is consistent and $S^j_{A\overline{B}}$ is inconsistent, then by Lemma 4 we have $\kappa^z_\Delta(AB) = 0$ and $\kappa^z_\Delta(A\overline{B}) = 1$ and thus $\kappa^z_\Delta(AB) < \kappa^z_\Delta(A\overline{B})$. In this case the checks in Lines 4 and 7 fail and *recZinf* returns "Yes".

In all other cases we have $\kappa^z_\Delta(AB) \not< \kappa^z_\Delta(A\overline{B})$ and *recZinf* returns "No" because one of the checks in Lines 4 and 7 succeeds.

In both cases the proposition holds.

Induction Step: Let $j > 0$ and assume the proposition holds for $j' = j - 1$. By assumption S^{j+1}_{AB} and $S^{j+1}_{A\overline{B}}$ are consistent, and thus, by Lemma 4 we have $\kappa^z_\Delta(AB) < j + 2$ and $\kappa^z_\Delta(A\overline{B}) < j + 2$. We can distinguish several cases

Case 1: S^j_{AB} is inconsistent
By Lemma 4 we have $\kappa^z_\Delta(AB) \geqslant j + 1$ implying that $\kappa^z_\Delta(AB) \not< \kappa^z_\Delta(A\overline{B})$. Also, the check in Line 4 succeeds and *recZinf* returns "No". The proposition holds.

Case 2: S^j_{AB} is consistent and S^j_{AB} is inconsistent
By Lemma 4 we have $\kappa^z_\Delta(AB) < j+1$ and $\kappa^z_\Delta(A\overline{B}) \geqslant j+1$ implying that $\kappa^z_\Delta(AB) < \kappa^z_\Delta(A\overline{B})$. The checks in Line 4 and 7 fail and *recZinf* returns "Yes". The proposition holds.

Case 3: S^j_{AB} is consistent and S^j_{AB} is consistent
In this case the function *recZinf* is called recursively for $j - 1$. The preconditions for applying this proposition for $j' = j - 1$ are satisfied, and therefore, by the induction hypothesis, *recZinf*(j) returns "Yes" iff $\kappa^z_\Delta(AB) < \kappa^z_\Delta(A\overline{B})$. The proposition holds. □

Using Proposition 3, we can now show the correctness of SZinf.

Theorem 2. *Given a consistent belief base Δ and $A, B \in \mathcal{L}_\Sigma$, the call* SZinf$(\Delta, A, B)$ *always terminates, and it returns "Yes" iff $A \mathrel{|\!\!\sim}^z_\Delta B$.*

Proof. For $A \equiv \bot$, SWinf(Δ, A, B) returns "Yes" (cf. Line 14) and $A \mathrel{|\!\!\sim}^z_\Delta B$, thus the theorem holds in this case. For the remainder of the proof assume that $A \not\equiv \bot$.

SWinf(Δ, A, B) terminates because for every recursive call of *recZinf* the index j is decreased by one, and the algorithm terminates at latest for $j = 0$. For $j = k$, the sets S^{j+1}_{AB} and $S^{j+1}_{A\overline{B}}$ are empty and are thus trivially consistent. By Proposition 3, the call SWinf(Δ, A, B) returns "Yes" iff $\kappa^z_\Delta(AB) < \kappa^z_\Delta(A\overline{B})$, which is equivalent to $A \mathrel{|\!\!\sim}^z_\Delta B$. □

8. Implementation and Evaluation Results

We implemented SWinf and SZinf in Python and using the SMT solver Z3 [28] accessed through the pySMT API [29]. In the implementation of SWinf, we used the optimizing features of the Z3 SMT Solver to find Pareto fronts [30] (which, in our case, is equivalent to finding the sets of all MSS and thus allows deriving the set of all MCS).

For checking the correctness of our implementations, we verified that the output of the implementations of SWinf and SZinf match the output of earlier implementations of system W [16] and of system Z [31]. For all queries with respect to all belief bases small enough to be processed by these older implementations without running into a timeout, the different implementations of system W and system Z, respectively, yielded exactly the same result.

For evaluating the implementations of SWinf and SZinf, belief bases and queries were constructed by a randomized scheme taking a signature Σ as input; a detailed description of this scheme and algorithms realizing it are given in [32]. Only consistent belief bases build during this process were used for benchmarking because system W as given in Definition 3 and system Z as given in Definition 6 are defined only for consistent belief bases; reasoning with system W with respect to belief bases that are only weakly consistent [33, 34] has been introduced only very recently [33, 24]. In the evaluation, belief bases with signature sizes $|\Sigma|$ ranging from 6 to 120 and number of conditionals $|\Delta|$ ranging from 6 to 200 were considered. For different $(|\Sigma|, |\Delta|)$-combinations, in summary, 2 800 belief bases and 28 000 queries were created; the belief bases and queries obtained thereby are available at the CLKR repository [35] at https://www.fernuni-hagen.de/wbs/clkr/ as *problem set* CLKR-PS004.

We benchmarked our (Partial Max-)SAT based Python implementations of SWinf and SZinf against existing Java-based approaches for inference according to system-W (WJ) and system-Z (ZJ) that both consider all possible worlds explicitly [16, 31]. No other previous system implementations of system W exist, and to the best of our knowledge all other existing implementations of system Z, e.g., [36, 37, 38], cannot handle belief bases over signature size of 50 or more.

Performance was assessed on a machine with an Intel i7-3770 CPU and 32GB RAM under Arch Linux (Linux kernel 6) and Python 3.11 (single threaded). Each implementation was tested on the full set of randomly generated belief bases and queries described above. The run times presented in Table 2 are averaged over 1,000 queries across 100 belief bases for every combination of $(|\Sigma|, |\Delta|)$. Time is presented in ms, timeout was set to 5 minutes.

The evaluation results in Table 2 show that only for the very smallest $(|\Sigma|, |\Delta|)$-combinations with $|\Sigma| = 6$ and $|\Delta| = 6$ the Java-based implementations WJ and ZJ of system W and system Z, respectively, are faster than our implementations of SWinf and SZinf. Furthermore, for every $(|\Sigma|, |\Delta|)$ combination with $|\Sigma| \geqslant 18$ both WJ and ZJ ran into a timeout in our evaluation scenario, while SWinf and SZinf successfully cope with all $(|\Sigma|, |\Delta|)$ combinations up to $|\Sigma| = 120$ and $|\Delta| = 200$. It is interesting to note that SZinfconsistently performs faster than SWinf in our evaluation only by a factor of up to 2 across all $(|\Sigma|, |\Delta|)$ combinations. In summary, this comparative evaluation showcases SWinf's and SZinf's superior scalability and ef-

$\|\Sigma\|$	6	8	10	12	14	16	18	20	30	40	50	60	60	60	60	80	80	80	80	100	100	100	100	120	120	120	120	120
$\|\Delta\|$	6	8	10	12	14	16	18	20	30	40	50	60	80	100	120	60	80	120	160	60	100	160	200	60	80	120	160	200
WJ	19	204	4967	248323	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.
SWinf	32	37	44	49	53	55	61	64	87	106	157	156	181	218	259	130	212	254	343	135	294	320	407	123	179	785	334	403
ZJ	11	52	302	1705	10337	80014	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.	to.
SZinf	32	35	38	40	42	44	45	46	55	62	74	78	106	129	143	78	95	150	200	75	104	175	219	75	92	128	177	224

Table 2
Evaluation of implementations of inference with system W and system Z, time in milliseconds, timeout (to.) is at 5 min.

ficiency, particularly in handling queries on larger belief bases within reasonable time frames.

9. Conclusions and Future Work

In this paper, we presented SAT and Partial MaxSAT based approaches for implementing nonmonotonic reasoning with system Z and system W. We presented the corresponding algorithms SZinf and SWinf and gave formal correctness proofs for them. The Python-based implementations of SZinf and SWinf use the power of current SAT and Partial MaxSAT solvers and scale up reasoning both with system Z and system W to a new dimension, easily coping with belief bases over 120 variables and containing up to 200 conditionals. This advancement also puts larger practical applications into reach for the first time.

Our current and future work includes extending the presented work in multiple ways. For instance, while the evaluation presented in Section 8 focuses on runtime, we will further evaluate SZinf and SWinf by taking also the memory consumption into account. We will also analyse the complexity of the algorithms. While previous practical applications were limited by small belief base sizes, we will address larger and more realistic scenarios in the medical and bio-medical domain as they have already been modelled with conditional logic [39], using the new opportunities opened up by the enriched power of our new implementations of nonmonotonic inference.

Acknowledgments

This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 512363537, grant BE 1700/12-1 awarded to Christoph Beierle.

References

[1] E. Adams, The Logic of Conditionals, D. Reidel, Dordrecht, 1975.

[2] D. Nute, Topics in Conditional Logic, D. Reidel Publishing Company, Dordrecht, Holland, 1980.

[3] W. Spohn, Ordinal conditional functions: a dynamic theory of epistemic states, in: W. Harper, B. Skyrms (Eds.), Causation in Decision, Belief Change, and Statistics, II, Kluwer Academic Publishers, 1988, pp. 105–134.

[4] S. Kraus, D. Lehmann, M. Magidor, Nonmonotonic reasoning, preferential models and cumulative logics, Artif. Intell. 44 (1990) 167–207.

[5] D. Dubois, H. Prade, Conditional objects as nonmonotonic consequence relationships, Special Issue on Conditional Event Algebra, IEEE Transactions on Systems, Man and Cybernetics 24 (1994) 1724–1740.

[6] S. Benferhat, D. Dubois, H. Prade, Nonmonotonic reasoning, conditional objects and possibility theory, Artificial Intelligence 92 (1997) 259–276.

[7] G. Kern-Isberner, Conditionals in nonmonotonic reasoning and belief revision, volume 2087 of *LNAI*, Springer, 2001.

[8] C. Beierle, C. Eichhorn, G. Kern-Isberner, S. Kutsch, Properties and interrelationships of skeptical, weakly skeptical, and credulous inference induced by classes of minimal models, Artificial Intelligence 297 (2021) 103489. doi:10.1016/j.artint.2021.103489.

[9] M. Goldszmidt, J. Pearl, Qualitative probabilities for default reasoning, belief revision, and causal modeling, Artificial Intelligence 84 (1996) 57–112.

[10] C. Komo, C. Beierle, Nonmonotonic reasoning from conditional knowledge bases with system W, Ann. Math. Artif. Intell. 90 (2022) 107–144. doi:10.1007/s10472-021-09777-9.

[11] D. Lehmann, M. Magidor, What does a conditional knowledge base entail?, Artif. Intell. 55 (1992) 1–60.

[12] S. Benferhat, C. Cayrol, D. Dubois, J. Lang, H. Prade, Inconsistency Management and Prioritized Syntax-Based Entailment, in: Proc. IJCAI'93, volume 1, Morgan Kaufmann Publishers, San Francisco, CA, USA, 1993, pp. 640–647.

[13] G. Kern-Isberner, C. Beierle, G. Brewka, Syntax splitting = relevance + independence: New postulates for nonmonotonic reasoning from conditional belief bases, in: D. Calvanese, E. Erdem, M. Thielscher (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the 17th International Conference, KR 2020, IJCAI Organization, 2020, pp. 560–571. doi:10.24963/kr.2020/56.

[14] J. Haldimann, C. Beierle, Inference with system W satisfies syntax splitting, in: G. Kern-Isberner, G. Lakemeyer, T. Meyer (Eds.), Proceedings of the 19th International Conference on Principles of Knowledge Representation and Reasoning, KR 2022, Haifa, Israel. July 31 - August 5, 2022, 2022, pp. 405–409. doi:10.24963/kr.2022/41.

[15] J. Heyninck, G. Kern-Isberner, T. Meyer, J. P. Haldimann, C. Beierle, Conditional syntax splitting for non-monotonic inference operators, in: B. Williams, Y. Chen, J. Neville (Eds.), Proceedings of the 37th AAAI Conference on Artificial Intelligence, volume 37, 2023, pp. 6416–6424. doi:10.1609/aaai.v37i5.25789.

[16] C. Beierle, J. Haldimann, D. Kollar, K. Sauerwald,

L. Schwarzer, An implementation of nonmonotonic reasoning with system W, in: R. Bergmann, L. Malburg, S. C. Rodermund, I. J. Timm (Eds.), KI 2022: Advances in Artificial Intelligence - 45th German Conference on AI, Trier, Germany, September 19-23, 2022, Proceedings, volume 13404 of *LNCS*, Springer, 2022, pp. 1–8. doi:10.1007/978-3-031-15791-2_1.

[17] C. Beierle, A. Spang, J. Haldimann, A Partial MaxSAT approach to nonmonotonic reasoning with system W, in: Proceedings of the 37th International Florida Artificial Intelligence Research Society Conference, 2024. doi:10.32473/FLAIRS.37.1.135330.

[18] B. de Finetti, La prévision, ses lois logiques et ses sources subjectives, Ann. Inst. H. Poincaré 7 (1937) 1–68. Engl. transl. *Theory of Probability*, J. Wiley & Sons, 1974.

[19] C. Komo, C. Beierle, Nonmonotonic inferences with qualitative conditionals based on preferred structures on worlds, in: U. Schmid, F. Klügl, D. Wolter (Eds.), KI 2020: Advances in Artificial Intelligence - 43rd German Conference on AI, Bamberg, Germany, September 21-25, 2020, Proceedings, volume 12325 of *LNCS*, Springer, 2020, pp. 102–115. doi:10.1007/978-3-030-58285-2_8.

[20] J. Pearl, System Z: A natural ordering of defaults with tractable applications to nonmonotonic reasoning, in: Proc. of the 3rd Conf. on Theoretical Aspects of Reasoning About Knowledge (TARK'1990), Morgan Kaufmann Publ. Inc., San Francisco, CA, USA, 1990, pp. 121–135.

[21] C. Beierle, C. Eichhorn, G. Kern-Isberner, Skeptical inference based on c-representations and its characterization as a constraint satisfaction problem, in: M. Gyssens, G. Simari (Eds.), Foundations of Information and Knowledge Systems - 9th International Symposium, FoIKS 2016, Linz, Austria, March 7–11, 2016. Proceedings, volume 9616 of *LNCS*, Springer, 2016, pp. 65–82. doi:10.1007/978-3-319-30024-5_4.

[22] J. Haldimann, C. Beierle, Properties of system W and its relationships to other inductive inference operators, in: I. Varzinczak (Ed.), Foundations of Information and Knowledge Systems - 12th International Symposium, FoIKS 2022, Helsinki, Finland, June 20-23, 2022, Proceedings, volume 13388 of *LNCS*, Springer, 2022, pp. 206–225. doi:10.1007/978-3-031-11321-5_12.

[23] J. Haldimann, C. Beierle, Characterizing multipreference closure with system W, in: F. D. de Saint-Cyr, M. Öztürk-Escoffier, N. Potyka (Eds.), Scalable Uncertainty Management - 15th International Conference, SUM 2022, Paris, France, October 17-19, 2022, Proceedings, volume 13562 of *LNCS*, Springer, 2022, pp. 79–91. doi:10.1007/978-3-031-18843-5_6.

[24] J. Haldimann, C. Beierle, Approximations of system W for inference from strongly and weakly consistent belief bases, Int. J. Approx. Reason. 175 (2024) 109295. doi:10.1016/j.ijar.2024.109295.

[25] J. Larrosa, E. Rollon, Towards a better understanding of (partial weighted) MaxSAT proof systems, in: L. Pulina, M. Seidl (Eds.), Theory and Applications of Satisfiability Testing - SAT 2020 - 23rd International Conference, Alghero, Italy, July 3-10, 2020, Proceedings, volume 12178 of *LNCS*, Springer, 2020, pp. 218–232. URL: https://doi.org/10.1007/978-3-030-51825-7_16.

[26] D. Lehmann, What does a conditional knowledge base entail?, in: R. J. Brachman, H. J. Levesque, R. Reiter (Eds.), Proceedings of the 1st International Conference on Principles of Knowledge Representation and Reasoning (KR'89). Toronto, Canada, May 15-18 1989, Morgan Kaufmann, 1989, pp. 212–222.

[27] M. Goldszmidt, J. Pearl, On the relation between rational closure and system-z, in: Proceedings of the Third International Workshop on Nonmonotonic Reasoning, May 31 – June 3, 1990, pp. 130–140.

[28] N. Bjørner, L. de Moura, L. Nachmanson, C. M. Wintersteiger, Programming Z3, Engineering Trustworthy Software Systems: 4th International School, SETSS 2018, Chongqing, China, April 7–12, 2018, Tutorial Lectures 4 (2019) 148–201.

[29] M. Gario, A. Micheli, Pysmt: a solver-agnostic library for fast prototyping of smt- based algorithms, in: SMT Workshop 2015, 2015.

[30] N. S. Bjørner, A. Phan, L. Fleckenstein, νz - an optimizing SMT solver, in: C. Baier, C. Tinelli (Eds.), Tools and Algorithms for the Construction and Analysis of Systems - 21st International Conference, TACAS 2015, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2015, London, UK, April 11-18, 2015. Proceedings, volume 9035 of *LNCS*, Springer, 2015, pp. 194–199. doi:10.1007/978-3-662-46681-0_14.

[31] S. Kutsch, C. Beierle, InfOCF-Web: An online tool for nonmonotonic reasoning with conditionals and ranking functions, in: Z. Zhou (Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, ijcai.org, 2021, pp. 4996–4999. doi:10.24963/ijcai.2021/711.

[32] M. von Berg, A. Sanin, C. Beierle, An implementation of nonmonotonic reasoning with c-representations using an SMT solver, Int. J. Approx. Reason. 175 (2024) 109285. doi:10.1016/j.ijar.2024.109285.

[33] J. Haldimann, C. Beierle, G. Kern-Isberner, T. Meyer, Conditionals, infeasible worlds, and reasoning with system W, The International FLAIRS Conference Proceedings 36 (2023). doi:10.32473/flairs.36.133268.

[34] J. P. Haldimann, T. Meyer, G. Kern-Isberner, C. Beierle, Rational closure extension in spo representable inductive inference operators, in: S. Gaggl, M. V. Martinez, M. Ortiz (Eds.), Logics in Artificial Intelligence 18th European Conference, JELIA 2023, Dresden, Germany, September 20–22, 2023, Proceedings, volume 12678 of *LNCS*, Springer, 2023, pp. 561–576. doi:10.1007/978-3-031-43619-2_38.

[35] C. Beierle, J. Haldimann, L. Schwarzer, CLKR – Conditional Logic and Knowledge Representation, KI – Künstliche Intelligenz (2024). URL: https://doi.org/10.1007/s13218-024-00842-z.

[36] M. Thimm, Tweety: A comprehensive collection of java libraries for logical aspects of artificial intelligence and knowledge representation, in: C. Baral, G. D. Giacomo, T. Eiter (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014, AAAI Press, 2014, pp. 528–537. URL: http://www.aaai.org/ocs/index.php/KR/KR14/paper/view/7811.

[37] C. Beierle, C. Eichhorn, S. Kutsch, A practical comparison of qualitative inferences with preferred ranking models, KI – Künstliche Intelligenz 31 (2017) 41–52. doi:10.1007/s13218-016-0453-9.

[38] A. Labecki, System Z default reasoning solver, 2017. URL: https://github.com/alabecki/System-Z, (accessed on 19/07(2924).

[39] J. Haldimann, A. Osiak, C. Beierle, Modelling and reasoning in biomedical applications with qualitative conditional logic, in: U. Schmid, F. Klügl, D. Wolter (Eds.), KI 2020: Advances in Artificial Intelligence - 43rd German Conference on AI, volume 12325 of *LNCS*, Springer, 2020, pp. 283–289. doi:10.1007/978-3-030-58285-2_24.