Towards a More Efficient Computation of Weighted Conditional Impacts for Relational Probabilistic Knowledge Bases Under Maximum Entropy Semantics

Marc Finthammer(✉) and Christoph Beierle

Department of Computer Science, FernUniversität in Hagen, Hagen, Germany
marc@finthammer.de

Abstract. While the complexity of the optimization problem to be solved when computing the Maximum Entropy distribution $P^*_{\mathcal{R}}$ of a knowledge base $\mathcal{R}$ grows dramatically when moving to the relational case, it has been shown that having the weighted conditional impacts (WCI) of $\mathcal{R}$ available, $P^*_{\mathcal{R}}$ can be computed much faster. Computing WCI in a straightforward manner readily gets infeasible due to the size of the set Ω of possible worlds. In this paper, we propose a new approach for computing the WCI without considering the worlds in Ω at all. We introduce the notion of *sat-pairs* and show how to determine the set $\mathcal{CSP}$ of all possible combinations of sat-pairs by employing combinatorial means. Using $\mathcal{CSP}$ instead of Ω for computing the WCI is a significant performance gain since $\mathcal{CSP}$ is typically much smaller than Ω. For a start, we focus on simple knowledge bases consisting of a single conditional. First evaluation results of an implemented algorithm illustrate the benefits of our approach.

1 Introduction

There is a long tradition of enriching propositional logic with probabilities ([8,17,19]), and relational probabilistic logics (e. g. [7,9,12,20,22]) provide a strong means to model uncertain knowledge about relations among individual objects. Here, we are especially interested in relational probabilistic conditionals.

Example 1. A movie actor can be awarded with certain awards (e. g. Oscar, Palme d'Or) and depending on that, a movie director might consider engaging that actor with a probability of 0.3. This scenario could be modeled by the relational probabilistic conditional $(considerEngagement(X, Z)|awardedWith(X, Y))[0.3]$ where the variable X stands for some actor, Y for some award, and Z for some movie director.

A set of relational probabilistic conditionals is called a knowledge base $\mathcal{R}$ and there generally exist many probability distributions which satisfy $\mathcal{R}$. Recently, several semantics for relational probabilistic conditionals have been introduced which employ the principle of maximum entropy (ME) to select the distribution

S. Hölldobler et al. (Eds.): KI 2015, LNAI 9324, pp. 72–86, 2015.
DOI: 10.1007/978-3-319-24489-1_6

which represents $\mathcal{R}$ in the most unbiased way, i. e. by adding as little information as possible (cf. [10,11,18,23]). Computing the ME distribution $P^*_{\mathcal{R}}$ of $\mathcal{R}$ requires solving an optimization problem whose complexity grows dramatically when moving from a propositional to a first-order setting. In [4], the well-known technique of generalized iterative scaling (GIS) [2] is used to develop an algorithm computing the ME distribution $P^*_{\mathcal{R}}$ under aggregating semantics [13] for the relational case. In [5], it is shown that $P^*_{\mathcal{R}}$ can be computed much faster by an algorithm which works on the so-called *weighted conditional impacts* (WCI) of $\mathcal{R}$ instead of the exponentially large set Ω of possible worlds. Equivalence classes of worlds are used for probabilistic logics, e. g. in [6,10,12,15,21], and WCI provide a more abstract view on these equivalence classes and their cardinalities. That way, WCI comprise the essential information about worlds and their interaction with the logical part of the conditionals in $\mathcal{R}$ in a condensed form. Thus, $P^*_{\mathcal{R}}$ can be computed more efficiently once the WCI have been determined. Since the WCI do not consider the given probabilities of the conditionals in $\mathcal{R}$, the WCI can also be reused when the probabilities in $\mathcal{R}$ are changed. However, computing the WCI straightforwardly requires to consider each world in Ω once which readily becomes infeasible due to the size of Ω.

In this paper, we propose a new approach for computing the WCI without considering the worlds in Ω at all. This approach abstracts from concrete worlds and from the concrete ground atoms satisfied by worlds. Instead, we focus on the possible numbers of ground atoms which in principle can be satisfied by some worlds. So we do not care which worlds cause a certain number of satisfied ground atoms, but we just make sure that we determine the particular numbers which are actually possible. We introduce the concept of *sat-pairs*, i. e. pairs of numbers which represent satisfiable numbers of ground atoms. We extend that concept to *combinations of sat-pairs* and show that the set $\mathcal{CSP}$, consisting of all possible combinations of sat-pairs and generally being much smaller then Ω, is a viable replacement for the set Ω. As a start, we focus on simple knowledge bases consisting of a single conditional, and point out possible extensions.

Section 2 briefly recalls the required background, Section 3 analyzes the WCI of atomic conditionals, Section 4 shows how to compute the WCI combinatorially, Section 5 presents the results of a practical algorithm, and Section 6 concludes.

2 Background

Relational Probabilistic Conditional Logic. Let $\mathcal{L}$ be a quantifier-free first-order language defined over a many-sorted first-order signature $\Sigma = (Sort, Const, Pred)$, where $Sort$ is a set of sorts, $Const$ is a finite set of constants, and $Pred$ a set of predicates. The language $(\mathcal{L}|\mathcal{L})^{prob}$ consists of *probabilistic conditionals* of the form $(B(\boldsymbol{X})|A(\boldsymbol{X}))[d]$ with $\boldsymbol{X}$ containing the variables of the formulas A and B, and where $d \in [0,1]$ is a probability; A finite set $\mathcal{R} \subseteq (\mathcal{L}|\mathcal{L})^{prob}$ is called a *knowledge base*; we always implicitly consider $\mathcal{R}$ together with some appropriate signature Σ.

$\mathcal{H}$ denotes the *Herbrand base*, i. e. the set containing all ground atoms over Σ, and $\Omega = \mathfrak{P}(\mathcal{H})$ is the set of all possible worlds (i. e. *Herbrand interpretations*), where $\mathfrak{P}$ is the power set operator. The satisfaction relation between a world $\omega \in \Omega$ and a ground atom at is defied as $\omega \models at$ *iff* $at \in \omega$ and extended to ground formulas in the usual way. $\Theta(\mathcal{V})$ denotes the set of all ground substitutions w. r. t. a set of variables $\mathcal{V}$. $A(\boldsymbol{a})$ denotes a ground instance of $A(X)$, where $\boldsymbol{a}$ contains the particular constants which substitute the variables in $\boldsymbol{X}$. The expression $\mathrm{gnd}(r)$ denotes the set of ground instances of a conditional $r = (B(\boldsymbol{X})|A(\boldsymbol{X}))[d]$, and we write $r(\boldsymbol{a})$ for a ground instance $(B(\boldsymbol{a})|A(\boldsymbol{a}))[d] \in \mathrm{gnd}(r)$. The *probabilistic interpretations* for $(\mathcal{L}|\mathcal{L})^{prob}$ are given by the set *Prob* of all probability distributions $P : \Omega \to [0,1]$ over possible worlds. P is extended to ground formulas $A(\boldsymbol{a})$ by defining $P(A(\boldsymbol{a})) := \sum_{\omega \models A(\boldsymbol{a})} P(\omega)$. The *aggregation semantics* [13] extends P to conditionals and resembles the definition of a conditional probability by summing up the probabilities of all respective ground formulas; it defines the satisfaction relation $\models_{\odot}$ for $r = (B(\boldsymbol{X})|A(\boldsymbol{X}))[d]$ by

$$P \models_{\odot} r \quad \textit{iff} \quad \frac{\sum_{r(\boldsymbol{a}) \in \mathrm{gnd}(r)} P\left(A(\boldsymbol{a})B(\boldsymbol{a})\right)}{\sum_{r(\boldsymbol{a}) \in \mathrm{gnd}(r)} P\left(A(\boldsymbol{a})\right)} = d \tag{1}$$

where $\sum_{r(\boldsymbol{a}) \in \mathrm{gnd}(r)} P\left(A(\boldsymbol{a})\right) > 0$. If $P \models_{\odot} r$ holds, we say that P *satisfies* r or P *is a model of* r. P *satisfies* a set of conditionals $\mathcal{R}$ if it satisfies every element of $\mathcal{R}$.

The principle of *maximum entropy* (*ME*) chooses the distribution P where the entropy $H(P)$ is maximal among all distributions satisfying $\mathcal{R}$ [10,18]. The ME model $P^*_{\mathcal{R}}$ for $\mathcal{R}$ based on aggregation semantics is uniquely defined [13] by the solution of the convex optimization problem $P^*_{\mathcal{R}} := \arg\max_{P \in Prob : P \models_{\odot} \mathcal{R}} H(P)$.

Weighted Conditional Impacts. For r: $(B(\boldsymbol{X})|A(\boldsymbol{X}))$, the *counting functions* (cf. [4,14]) $\mathrm{app}, \mathrm{ver} : \Omega \to \mathbb{N}_0$ are:

$$\mathrm{app}(\omega) := \left|\left\{r(\boldsymbol{a}) \in \mathrm{gnd}(r) \mid \omega \models A(\boldsymbol{a})\right\}\right| \tag{2}$$

$$\mathrm{ver}(\omega) := \left|\left\{r(\boldsymbol{a}) \in \mathrm{gnd}(r) \mid \omega \models A(\boldsymbol{a})B(\boldsymbol{a})\right\}\right| \tag{3}$$

For $\omega \in \Omega$, $\mathrm{app}(\omega)$ yields the number of ground instances of r which are *applicable* w. r. t. ω, and $\mathrm{ver}(\omega)$ yields the number of ground instances of r *verified* by ω.

Definition 1 (va-Pair). *The function* $\mathrm{va}_{\Omega} : \Omega \to \mathbb{N}_0 \times \mathbb{N}_0$ *with*

$$\mathrm{va}_{\Omega}(\omega) := \left\langle \mathrm{ver}(\omega), \mathrm{app}(\omega) \right\rangle \tag{4}$$

w. r. t. a conditional r *is called the* va-pair function *of* r. *A function value* $\mathrm{va}_{\Omega}(\omega)$ *is called the* va-pair *of* r *with respect to* ω. *The image of* va_{Ω} *is denoted by*

$$\mathcal{VA} := \{\langle v, a\rangle \in \mathbb{N}_0 \times \mathbb{N}_0 \mid \langle v, a\rangle = \mathrm{va}_{\Omega}(\omega),\ \omega \in \Omega\} \tag{5}$$

and called the set of va-pairs *of* r, *and a pair* $\langle v, a\rangle \in \mathcal{VA}$ *is called a* va-pair of r.

For example, if for some $\omega \in \Omega$, ten ground instances of a conditional r are applicable and six ground instances are verified, then we have $\mathrm{app}(\omega) = 10$ and $\mathrm{ver}(\omega) = 6$ and consequently $\langle 6, 10\rangle$ is a va-pair of r.

Definition 2 (Conditional Impact). *The function* $\gamma_{\mathcal{R}} : \Omega \to (\mathbb{N}_0 \times \mathbb{N}_0)^m$ *with*

$$\gamma_{\mathcal{R}}(\omega) := \big[\mathrm{va}_{\Omega,1}(\omega), \ldots, \mathrm{va}_{\Omega,m}(\omega)\big] \tag{6}$$

is called the conditional impact function *of* $\mathcal{R}$. *The value* $\gamma_{\mathcal{R}}(\omega)$ *is called the* conditional impact caused by ω *on the ground instances of* $\mathcal{R}$. *The image of* $\gamma_{\mathcal{R}}$ *is denoted by*

$$\Gamma_{\mathcal{R}} := \gamma_{\mathcal{R}}(\Omega) = \{\boldsymbol{\gamma} \in (\mathbb{N}_0 \times \mathbb{N}_0)^m \mid \boldsymbol{\gamma} = \gamma_{\mathcal{R}}(\omega) \text{ for some } \omega \in \Omega\} \tag{7}$$

and called the set of conditional impacts *of* $\mathcal{R}$,
and a tuple $\boldsymbol{\gamma} = \big[\langle v_1, a_1\rangle, \ldots, \langle v_m, a_m\rangle\big] \in \Gamma_{\mathcal{R}}$ *is called a* conditional impact of $\mathcal{R}$.

For example, a conditional impact of a set $\mathcal{R}$ consisting of 4 conditionals could be $\boldsymbol{\gamma} = \big[\langle 6, 10\rangle, \langle 1, 8\rangle, \langle 0, 6\rangle, \langle 5, 9\rangle\big]$. Conditional impacts were introduced by Kern-Isberner [10] as *conditional structures* in a propositional setting, using a free Abelian group construction. In the propositional case, each va-pair is $\langle 0, 0\rangle$, $\langle 0, 1\rangle$, or $\langle 1, 1\rangle$.

Definition 3 (WCI). *The function* $\mathrm{wgt}_{\mathcal{R}} : \Gamma_{\mathcal{R}} \to \mathbb{N}_0$ *with*

$$\mathrm{wgt}_{\mathcal{R}}(\boldsymbol{\gamma}) := \big|\gamma_{\mathcal{R}}^{-1}(\boldsymbol{\gamma})\big| = \big|\{\omega \in \Omega \mid \gamma_{\mathcal{R}}(\omega) = \boldsymbol{\gamma}\}\big|$$

is called the weighting function of $\mathcal{R}$. *The pair* $(\Gamma_{\mathcal{R}}, \mathrm{wgt}_{\mathcal{R}})$ *is called the* **w**eighted **c**onditional **i**mpacts (WCI) *of* $\mathcal{R}$.

Proposition 1 ([5]). *The ME-distribution* $P^*_{\mathcal{R}}$ *can be computed by a GIS-algorithm which solely works on the WCI of* $\mathcal{R}$ *and does not refer to* Ω *at all.*

ME-Computation on Weighted Conditional Impacts. Since the WCI are in general much smaller than Ω, the adopted GIS-algorithm computes the ME-distribution $P^*_{\mathcal{R}}$ much faster and requires much less space than a comparable algorithm working on Ω directly. So the WCI comprise the essential information about the qualitative part of $\mathcal{R}$ and the worlds in Ω in a most condensed form. Once determined, the WCI can also be reused together with different probabilities for the conditionals in $\mathcal{R}$. A drawback of the generic algorithm $\texttt{WCI}_{\mathrm{gen}}$ [5] computing the WCI is that is has to consider each world from Ω once for determining the WCI. However, note that once the WCI are available, the set Ω does not have to be considered anymore in the whole process of determining the ME-distribution $P^*_{\mathcal{R}}$ according to Prop. 1. Thus, our objective is to a develop a more efficient approach to compute the WCI without considering the set Ω at all.

3 Basic Case: WCI of a Single Conditional

Since analyzing the WCI for a whole set $\mathcal{R}$ of m conditionals is a much too complex task to begin with, we focus on the WCI of single conditional, i. e. we consider a set $\mathcal{R} = \{r\}$ throughout the rest of this paper. That way, we do not have to consider conditional impacts in terms of m-tuples of va-pairs as $\gamma_{\mathcal{R}}$ in Def. 2, but we can just consider the va-pairs of r themselves. That is, the set of conditional impacts under consideration is $\Gamma_{\mathcal{R}} = \{[\langle v, a\rangle] \mid \langle v, a\rangle \in \mathcal{VA}\}$ with $\mathcal{VA}$ being the set of va-pairs of r. Analogously to $\mathrm{wgt}_{\mathcal{R}}$ (Def. 3), the function $\mathrm{wgt} : \mathcal{VA} \to \mathbb{N}$ with

$$\mathrm{wgt}(\langle v, a\rangle) := \mathrm{wgt}_{\mathcal{R}}([\langle v, a\rangle]) = |\mathrm{va}_{\Omega}^{-1}(\langle v, a\rangle)| \tag{8}$$

is the weighting function of r. This allows us to denote the WCI of a single conditional r more directly as $(\mathcal{VA}, \mathrm{wgt})$ when investigating va-Pairs of r and their weights in the following. Since we will often refer to the number of ground instances of r, we define the compact notation $G := |\,\mathrm{gnd}(r)|$.

Note that when discussing values depending on r, the values of $\mathcal{VA}$, wgt, G, etc. also depend on the number of constants in the given signature Σ.

Proposition 2 (Upper Bound for $|\mathcal{VA}|$). *The number of va-pairs of a conditional is bounded by its number of ground instances by $|\mathcal{VA}| \leq va_{\lim}(G)$ with:*

$$va_{\lim}(G) = \sum_{a=0}^{G}\sum_{v=0}^{a} 1 = \sum_{a=0}^{G} a + 1 = \sum_{a=1}^{G+1} a = \frac{(G+1)\cdot(G+2)}{2} = \frac{G^2+3G+2}{2} \tag{9}$$

For instance, a conditional with 20 ground instances can have at most 231 va-pairs.

Atomic Conditionals. Since the set $\mathrm{gnd}(r)$ plays an important role when considering the constitution of the set $\mathcal{VA}$ of a conditional r, we want to take a closer look at the syntactical structure of the ground instances arising from r. Here, we focus on a simple syntactical structure, already covering many relevant aspects:

Definition 4 (Atomic Conditional). *Let* Cons *and* Ante *be atoms of different predicates. Then $r = (\,\mathsf{Cons} \mid \mathsf{Ante}\,)$ is called an* atomic conditional.

The atoms Ante and Cons can have an arbitrary number of arguments, being either a variable or a constant. For the rest of this paper, we focus on an atomic conditional r with only variables, since constants do not have any further effect when constructing the ground instances of $\mathcal{R} = \{r\}$.

Example 2 (Atomic Conditionals). Some examples for atomic conditionals are

$$r_{X|X} : (c(X)|a(X)) \quad r_{X|XY} : (c(X)|a(X,Y))$$
$$r_{Z|Y} : (c(Z)|a(Y)) \quad r_{XZ|X} : (c(X,Z)|a(X)) \quad r_{XZ|XY} : (c(X,Z)|a(X,Y))$$

Note that the conditional given in Ex. 1 corresponds to the schema used in $r_{XZ|XY}$.

The above atomic conditionals just differ in the numbers and positions of their variables. We introduce the following notation to refer to the respective sets of variables of an atomic conditional $r = (\,\mathsf{Cons} \mid \mathsf{Ante}\,)$:

- $\mathrm{C} \cap \mathrm{A}$: variables appearing in both Cons and Ante
- $\mathrm{A} \backslash \mathrm{C}$: variables appearing exclusively in Ante
- $\mathrm{C} \backslash \mathrm{A}$: variables appearing exclusively in Cons

Accordingly, with $\Theta_{\mathrm{C}\cap\mathrm{A}}(r)$, $\Theta_{\mathrm{A}\backslash\mathrm{C}}(r)$, and $\Theta_{\mathrm{C}\backslash\mathrm{A}}(r)$ we denote the corresponding sets of substitutions with respect to the particular sets of variables, and with

$$G_{\mathrm{C}\cap\mathrm{A}} := |\Theta_{\mathrm{C}\cap\mathrm{A}}(r)|, \quad G_{\mathrm{A}\backslash\mathrm{C}} := |\Theta_{\mathrm{A}\backslash\mathrm{C}}(r)|, \quad G_{\mathrm{C}\backslash\mathrm{A}} := |\Theta_{\mathrm{C}\backslash\mathrm{A}}(r)|$$

we denote the respective number of substitutions. Since $\Theta_{\mathrm{C}\cap\mathrm{A}}(r) \cup \Theta_{\mathrm{A}\backslash\mathrm{C}}(r) \cup \Theta_{\mathrm{C}\backslash\mathrm{A}}(r) = \Theta(r)$ and since the three sets of substitutions are pairwise disjoint, we have $G = G_{\mathrm{C}\cap\mathrm{A}} \cdot G_{\mathrm{A}\backslash\mathrm{C}} \cdot G_{\mathrm{C}\backslash\mathrm{A}}$. Note that each of the sets $\mathrm{C}\cap\mathrm{A}$, $\mathrm{A}\backslash\mathrm{C}$, and $\mathrm{C}\backslash\mathrm{A}$ may be empty for a particular atomic conditional; however $G_{\mathrm{C}\cap\mathrm{A}} \geq 1$, $G_{\mathrm{A}\backslash\mathrm{C}} \geq 1$, and $G_{\mathrm{C}\backslash\mathrm{A}} \geq 1$ always hold, since $\Theta(\emptyset)$ contains the empty substitution. The next example illustrates these definitions by an atomic conditional $r_{X_s Z_u | X_s Y_t}$ covering all three kinds of appearances of variables (cf. Ex. 1 for some practical interpretation of such a conditional):

Example 3. For $\Sigma = (Sort, Const, Pred)$ with $Sort = \{S, T, U\}$, $Pred = \{a/(S,T),\ c/(S,U)\}$, and $Const = Const_{(S)} \cup Const_{(T)} \cup Const_{(U)}$ with

$$Const_{(S)} = \{s_1, \ldots, s_3\},\ Const_{(T)} = \{t_1, \ldots, t_4\},\ Const_{(U)} = \{u_1, \ldots, u_5\}$$

together with the conditional $r_{X_s Z_u | X_s Y_t}\colon\ (c(X,Z) | a(X,Y))$ we have:

$$\begin{aligned}
\mathrm{C}\cap\mathrm{A} = \{X\}, \quad \mathrm{A}\backslash\mathrm{C} = \{Y\}, \quad \mathrm{C}\backslash\mathrm{A} &= \{Z\} \\
\Theta_{\mathrm{C}\cap\mathrm{A}}(r_{X_s Z_u | X_s Y_t}) &= \{\{X/s_1\}, \{X/s_2\}, \{X/s_3\}\} \\
\Theta_{\mathrm{A}\backslash\mathrm{C}}(r_{X_s Z_u | X_s Y_t}) &= \{\{Y/t_1\}, \{Y/t_2\}, \{Y/t_3\}, \{Y/t_4\}\} \\
\Theta_{\mathrm{C}\backslash\mathrm{A}}(r_{X_s Z_u | X_s Y_t}) &= \{\{Z/u_1\}, \{Z/u_2\}, \{Z/u_3\}, \{Z/u_4\}, \{Z/u_5\}\} \\
G_{\mathrm{C}\cap\mathrm{A}} = |\Theta_{\mathrm{C}\cap\mathrm{A}}(r_{X_s Z_u | X_s Y_t})| &= |\,Const_{(S)}\,| = 3 \\
G_{\mathrm{A}\backslash\mathrm{C}} = |\Theta_{\mathrm{A}\backslash\mathrm{C}}(r_{X_s Z_u | X_s Y_t})\ | &= |\,Const_{(T)}| = 4 \\
G_{\mathrm{C}\backslash\mathrm{A}} = |\Theta_{\mathrm{C}\backslash\mathrm{A}}(r_{X_s Z_u | X_s Y_t})\,| &= |\,Const_{(U)}| = 5
\end{aligned}$$

The total number of ground instances is $G = G_{\mathrm{C}\cap\mathrm{A}} \cdot G_{\mathrm{A}\backslash\mathrm{C}} \cdot G_{\mathrm{C}\backslash\mathrm{A}} = 3 \cdot 4 \cdot 5 = 60$; all these ground instances are depicted in Tab. 1. According to (9), an upper bound for number of va-pairs is $|\mathcal{VA}| \leq va_{\lim}(60) = 1{,}891$. Computing the set $\mathcal{VA}$ gives us the actual size $|\mathcal{VA}| = 348$. However, the number of worlds is:

$$|\Omega| = 2^{|\mathcal{H}|} = 2^{3\cdot 4 + 3\cdot 5} = 2^{27} = 134{,}217{,}728$$

So the set $\mathcal{VA}$, which emerges from Ω according to (5), is indeed much smaller than the original set Ω:

$$|\Omega| = 134{,}217{,}728 \ \gg\ |\mathcal{VA}| = 348$$

Table 1. Complete ground-instance-table of $r_{X_s Z_u | X_s Y_t}$

$c(s_1,Z) \mid a(s_1,Y)$	$c(s_2,Z) \mid a(s_2,Y)$	$c(s_3,Z) \mid a(s_3,Y)$
$c(s_1,u_1) \mid a(s_1,t_1)$	$c(s_2,u_1) \mid a(s_2,t_1)$	$c(s_3,u_1) \mid a(s_3,t_1)$
$c(s_1,u_2) \mid a(s_1,t_1)$	$c(s_2,u_2) \mid a(s_2,t_1)$	$c(s_3,u_2) \mid a(s_3,t_1)$
$c(s_1,u_3) \mid a(s_1,t_1)$	$c(s_2,u_3) \mid a(s_2,t_1)$	$c(s_3,u_3) \mid a(s_3,t_1)$
$c(s_1,u_4) \mid a(s_1,t_1)$	$c(s_2,u_4) \mid a(s_2,t_1)$	$c(s_3,u_4) \mid a(s_3,t_1)$
$c(s_1,u_5) \mid a(s_1,t_1)$	$c(s_2,u_5) \mid a(s_2,t_1)$	$c(s_3,u_5) \mid a(s_3,t_1)$
$c(s_1,u_1) \mid a(s_1,t_2)$	$c(s_2,u_1) \mid a(s_2,t_2)$	$c(s_3,u_1) \mid a(s_3,t_2)$
$c(s_1,u_2) \mid a(s_1,t_2)$	$c(s_2,u_2) \mid a(s_2,t_2)$	$c(s_3,u_2) \mid a(s_3,t_2)$
$c(s_1,u_3) \mid a(s_1,t_2)$	$c(s_2,u_3) \mid a(s_2,t_2)$	$c(s_3,u_3) \mid a(s_3,t_2)$
$c(s_1,u_4) \mid a(s_1,t_2)$	$c(s_2,u_4) \mid a(s_2,t_2)$	$c(s_3,u_4) \mid a(s_3,t_2)$
$c(s_1,u_5) \mid a(s_1,t_2)$	$c(s_2,u_5) \mid a(s_2,t_2)$	$c(s_3,u_5) \mid a(s_3,t_2)$
$c(s_1,u_1) \mid a(s_1,t_3)$	$c(s_2,u_1) \mid a(s_2,t_3)$	$c(s_3,u_1) \mid a(s_3,t_3)$
$c(s_1,u_2) \mid a(s_1,t_3)$	$c(s_2,u_2) \mid a(s_2,t_3)$	$c(s_3,u_2) \mid a(s_3,t_3)$
$c(s_1,u_3) \mid a(s_1,t_3)$	$c(s_2,u_3) \mid a(s_2,t_3)$	$c(s_3,u_3) \mid a(s_3,t_3)$
$c(s_1,u_4) \mid a(s_1,t_3)$	$c(s_2,u_4) \mid a(s_2,t_3)$	$c(s_3,u_4) \mid a(s_3,t_3)$
$c(s_1,u_5) \mid a(s_1,t_3)$	$c(s_2,u_5) \mid a(s_2,t_3)$	$c(s_3,u_5) \mid a(s_3,t_3)$
$c(s_1,u_1) \mid a(s_1,t_4)$	$c(s_2,u_1) \mid a(s_2,t_4)$	$c(s_3,u_1) \mid a(s_3,t_4)$
$c(s_1,u_2) \mid a(s_1,t_4)$	$c(s_2,u_2) \mid a(s_2,t_4)$	$c(s_3,u_2) \mid a(s_3,t_4)$
$c(s_1,u_3) \mid a(s_1,t_4)$	$c(s_2,u_3) \mid a(s_2,t_4)$	$c(s_3,u_3) \mid a(s_3,t_4)$
$c(s_1,u_4) \mid a(s_1,t_4)$	$c(s_2,u_4) \mid a(s_2,t_4)$	$c(s_3,u_4) \mid a(s_3,t_4)$
$c(s_1,u_5) \mid a(s_1,t_4)$	$c(s_2,u_5) \mid a(s_2,t_4)$	$c(s_3,u_5) \mid a(s_3,t_4)$

Syntactical Structure of Ground Instances. In the following, we employ Ex. 2 to explain the general concepts. The tabular representation of the ground instances of $r_{X_s Z_u | X_s Y_t}$ in Tab. 1 is called the *ground-instance-table* of $r_{X_s Z_u | X_s Y_t}$ and consists of $3 = G_{\mathrm{C} \cap \mathrm{A}}$ sub-tables. Each of these sub-tables emerges from one particular substitution for the variable $X \in \mathrm{C} \cap \mathrm{A}$ (cf. the respective table-headers).

The ground atoms appearing in each sub-table are pairwise disjoint due to the different substitutions for X, and our results will apply to each of the sub-tables in the same way. Thus, we continue our analysis with the first sub-table, which contains all ground instances emerging from the substitution X/s_1; we refer to that table as the s_1-table. The ground instances in the sub-table are divided into $4 = G_{\mathrm{A} \setminus \mathrm{C}}$ blocks (represented by horizontal lines), whereas each block considers a particular substitution for the variable $Y \in \mathrm{A} \setminus \mathrm{C}$. Finally, each block contains $5 = G_{\mathrm{C} \setminus \mathrm{A}}$ ground instances emerging from the substitutions for the variable $Z \in \mathrm{C} \setminus \mathrm{A}$.

Note that, apart from the concrete example, the ground-instance-table of every atomic conditional has such a three-leveled block-structure, which is a direct consequence of the three sets $\mathrm{C} \cap \mathrm{A}$, $\mathrm{A} \setminus \mathrm{C}$, and $\mathrm{C} \setminus \mathrm{A}$. In particular, this also holds for any atomic conditional with more than three variables.

Table 2. General block-structure of a sub-table

$c(s_1,Z)$	$a(s_1,Y)$
$c(s_1,u_1)$	$a(s_1,t_1)$
$c(s_1,u_2)$	$a(s_1,t_1)$
$c(s_1,u_3)$	$a(s_1,t_1)$
$c(s_1,u_4)$	$a(s_1,t_1)$
$c(s_1,u_5)$	$a(s_1,t_1)$
$c(s_1,u_1)$	$a(s_1,t_2)$
$c(s_1,u_2)$	$a(s_1,t_2)$
$c(s_1,u_3)$	$a(s_1,t_2)$
$c(s_1,u_4)$	$a(s_1,t_2)$
$c(s_1,u_5)$	$a(s_1,t_2)$
$c(s_1,u_1)$	$a(s_1,t_3)$
$c(s_1,u_2)$	$a(s_1,t_3)$
$c(s_1,u_3)$	$a(s_1,t_3)$
$c(s_1,u_4)$	$a(s_1,t_3)$
$c(s_1,u_5)$	$a(s_1,t_3)$
$c(s_1,u_1)$	$a(s_1,t_4)$
$c(s_1,u_2)$	$a(s_1,t_4)$
$c(s_1,u_3)$	$a(s_1,t_4)$
$c(s_1,u_4)$	$a(s_1,t_4)$
$c(s_1,u_5)$	$a(s_1,t_4)$

Table 3. Atoms satisfied by ω'

$c(s_1,Z)$	$a(s_1,Y)$
$c(s_1,u_1)$	$a(s_1,t_1)$
$c(s_1,u_2)$	$a(s_1,t_1)$
$c(s_1,u_3)$	$a(s_1,t_1)$
$c(s_1,u_4)$	$a(s_1,t_1)$
$c(s_1,u_5)$	$a(s_1,t_1)$
$c(s_1,u_1)$	$a(s_1,t_2)$
$c(s_1,u_2)$	$a(s_1,t_2)$
$c(s_1,u_3)$	$a(s_1,t_2)$
$c(s_1,u_4)$	$a(s_1,t_2)$
$c(s_1,u_5)$	$a(s_1,t_2)$
$c(s_1,u_1)$	$a(s_1,t_3)$
$c(s_1,u_2)$	$a(s_1,t_3)$
$c(s_1,u_3)$	$a(s_1,t_3)$
$c(s_1,u_4)$	$a(s_1,t_3)$
$c(s_1,u_5)$	$a(s_1,t_3)$
$c(s_1,u_1)$	$a(s_1,t_4)$
$c(s_1,u_2)$	$a(s_1,t_4)$
$c(s_1,u_3)$	$a(s_1,t_4)$
$c(s_1,u_4)$	$a(s_1,t_4)$
$c(s_1,u_5)$	$a(s_1,t_4)$

Table 4. Atoms satisfied by ω''

$c(s_1,Z)$	$a(s_1,Y)$
$c(s_1,u_1)$	$a(s_1,t_1)$
$c(s_1,u_2)$	$a(s_1,t_1)$
$c(s_1,u_3)$	$a(s_1,t_1)$
$c(s_1,u_4)$	$a(s_1,t_1)$
$c(s_1,u_5)$	$a(s_1,t_1)$
$c(s_1,u_1)$	$a(s_1,t_2)$
$c(s_1,u_2)$	$a(s_1,t_2)$
$c(s_1,u_3)$	$a(s_1,t_2)$
$c(s_1,u_4)$	$a(s_1,t_2)$
$c(s_1,u_5)$	$a(s_1,t_2)$
$c(s_1,u_1)$	$a(s_1,t_3)$
$c(s_1,u_2)$	$a(s_1,t_3)$
$c(s_1,u_3)$	$a(s_1,t_3)$
$c(s_1,u_4)$	$a(s_1,t_3)$
$c(s_1,u_5)$	$a(s_1,t_3)$
$c(s_1,u_1)$	$a(s_1,t_4)$
$c(s_1,u_2)$	$a(s_1,t_4)$
$c(s_1,u_3)$	$a(s_1,t_4)$
$c(s_1,u_4)$	$a(s_1,t_4)$
$c(s_1,u_5)$	$a(s_1,t_4)$

Table 2 shows the s_1-table, where identical ground atoms in the antecedence and consequence, respectively, are highlighted accordingly. Since the atoms in the antecedence and consequence of an atomic conditional must be of different predicates, the respective sets of ground atoms are always disjoint. So there are $4 = G_{\mathrm{A}\backslash\mathrm{C}}$ different antecedence ground atoms (*a-atoms*) and $5 = G_{\mathrm{C}\backslash\mathrm{A}}$ different consequence ground atoms (*c-atoms*) in the sub-table.

Numbers of Satisfied Ground Atoms. For some arbitrary world, let $satC$ and $satA$ denote the number of c-atoms and a-atoms, respectively, in a sub-table of an atomic conditional which are satisfied by that world. Then $satC \in N_C = \{0, 1, \ldots, G_{\mathrm{C}\backslash\mathrm{A}}\}$ and $satA \in N_A = \{0, 1, \ldots, G_{\mathrm{A}\backslash\mathrm{C}}\}$ holds, i. e. the numbers of satisfied c-atoms and a-atoms are from the respective range. On the other hand, for every such pair

$$(satC, satA) \in \mathcal{SP} := N_C \times N_A,$$

called *sat-pair* of an atomic conditional, there exists a world satisfying the respective number of c-atoms and a-atoms. We illustrate these ideas by considering the world

$$\omega' = \{c(s_1, u_1), c(s_1, u_3), c(s_1, u_4), a(s_1, t_1), a(s_1, t_2)\}.$$

In Tab. 3, the $satC' = 3$ and $satA' = 2$ atoms satisfied by ω' are highlighted. So $(satC', satA') = (3, 2)$ is the corresponding sat-pair. Changing our view back to the ground instances in Tab. 3, we easily see that $\mathrm{app}(\omega') = G_{\mathrm{C}\setminus\mathrm{A}} \cdot satA' = 5 \cdot 2 = 10$ holds, i. e. 10 ground instances are applicable w. r. t. ω'. Furthermore, $\mathrm{ver}(\omega') = satC' \cdot satA' = 3 \cdot 2 = 6$ holds, i. e. 6 of these ground instances are also verified by ω'. So we have $\mathrm{va}_\Omega(\omega') = \langle \mathrm{ver}(\omega'), \mathrm{app}(\omega') \rangle = \langle 6, 10 \rangle \in \mathcal{VA}$. Next, we consider another world

$$\omega'' = \{c(s_1, u_2), c(s_1, u_4), c(s_1, u_5),\ a(s_1, t_1), a(s_1, t_4)\}.$$

Table 4 illustrates that ω'' satisfies some different ground atoms (compared to ω') and therefore also verifies some different ground instances. Nevertheless, ω' and ω'' coincide in the actual *numbers* of satisfied c-atoms and a-atoms, i. e. $(satC', satA') = (satC'', satA'') = (3, 2)$ holds, and therefore both worlds have the same va-pair, i. e. $\mathrm{va}_\Omega(\omega') = \mathrm{va}_\Omega(\omega'') = \langle 6, 10 \rangle$. This example illustrates that if we are interested in possible va-pairs, then we do not necessarily have to consider worlds, but we can consider sat-pairs instead.

4 Computing WCI Using Combinatorics

Next, we show how the above consideration concerning just one sub-table can be extended to the complete ground-instance-table. So we consider the world

$$\begin{aligned}\omega''' = \{&c(s_1, u_1), c(s_1, u_3), c(s_1, u_4), && a(s_1, t_1), a(s_1, t_2)\\ &c(s_2, u_2), c(s_2, u_3), c(s_2, u_4), && a(s_2, t_2), a(s_2, t_4)\\ &c(s_3, u_4), && a(s_3, t_1), a(s_3, t_2), a(s_3, t_3), a(s_3, t_4)\}\end{aligned}$$

Table 5 shows the complete ground-instance-table, where the atoms satisfied by ω''' are highlighted accordingly. With respect to the three sub-tables, we have the sat-pairs

$$(satC_1''', satA_1''') = (3, 2),\ \ (satC_2''', satA_2''') = (3, 2),\ \ (satC_3''', satA_3''') = (1, 4).$$

The overall number of applicable and verified ground instances can be determined by summing up the particular result of each sub-table:

$$\mathrm{app}(\omega''') = G_{\mathrm{C}\setminus\mathrm{A}} \cdot satA_1''' + G_{\mathrm{C}\setminus\mathrm{A}} \cdot satA_2''' + G_{\mathrm{C}\setminus\mathrm{A}} \cdot satA_3''' = 5 \cdot 2 + 5 \cdot 2 + 5 \cdot 4 = 40$$
$$\mathrm{ver}(\omega''') = satC_1''' \cdot satA_1''' + satC_2''' \cdot satA_2''' + satC_3''' \cdot satA_3''' = 3 \cdot 2 + 3 \cdot 2 + 1 \cdot 4 = 16$$

This yields $\mathrm{va}_\Omega(\omega''') = \langle \mathrm{ver}(\omega'''), \mathrm{app}(\omega''') \rangle = \langle 16, 40 \rangle \in \mathcal{VA}$.

In these computations, the numbers resulting from the particular sat-pairs are just summed up, hence it does not actually matter from which concrete sub-table a sat-pair arises. Thus, when considering several sat-pairs, their particular order does not matter, and it is sufficient to consider unordered combinations of sat-pairs. Thus, the above combination of sat-pairs can be represented by the multiset (denoted by double braces):

$$CSP''' = \{\!\{(3, 2), (3, 2), (1, 4)\}\!\} \tag{10}$$

Table 5. Atoms satisfied by ω''' in the ground-instance-table

$c(s_1,Z)$ | $a(s_1,Y)$	$c(s_2,Z)$ | $a(s_2,Y)$	$c(s_3,Z)$ | $a(s_3,Y)$
$c(s_1,u_1)$ | $a(s_1,t_1)$	$c(s_2,u_1)$ | $a(s_2,t_1)$	$c(s_3,u_1)$ | $a(s_3,t_1)$
$c(s_1,u_2)$ | $a(s_1,t_1)$	$c(s_2,u_2)$ | $a(s_2,t_1)$	$c(s_3,u_2)$ | $a(s_3,t_1)$
$c(s_1,u_3)$ | $a(s_1,t_1)$	$c(s_2,u_3)$ | $a(s_2,t_1)$	$c(s_3,u_3)$ | $a(s_3,t_1)$
$c(s_1,u_4)$ | $a(s_1,t_1)$	$c(s_2,u_4)$ | $a(s_2,t_1)$	$c(s_3,u_4)$ | $a(s_3,t_1)$
$c(s_1,u_5)$ | $a(s_1,t_1)$	$c(s_2,u_5)$ | $a(s_2,t_1)$	$c(s_3,u_5)$ | $a(s_3,t_1)$
$c(s_1,u_1)$ | $a(s_1,t_2)$	$c(s_2,u_1)$ | $a(s_2,t_2)$	$c(s_3,u_1)$ | $a(s_3,t_2)$
$c(s_1,u_2)$ | $a(s_1,t_2)$	$c(s_2,u_2)$ | $a(s_2,t_2)$	$c(s_3,u_2)$ | $a(s_3,t_2)$
$c(s_1,u_3)$ | $a(s_1,t_2)$	$c(s_2,u_3)$ | $a(s_2,t_2)$	$c(s_3,u_3)$ | $a(s_3,t_2)$
$c(s_1,u_4)$ | $a(s_1,t_2)$	$c(s_2,u_4)$ | $a(s_2,t_2)$	$c(s_3,u_4)$ | $a(s_3,t_2)$
$c(s_1,u_5)$ | $a(s_1,t_2)$	$c(s_2,u_5)$ | $a(s_2,t_2)$	$c(s_3,u_5)$ | $a(s_3,t_2)$
$c(s_1,u_1)$ | $a(s_1,t_3)$	$c(s_2,u_1)$ | $a(s_2,t_3)$	$c(s_3,u_1)$ | $a(s_3,t_3)$
$c(s_1,u_2)$ | $a(s_1,t_3)$	$c(s_2,u_2)$ | $a(s_2,t_3)$	$c(s_3,u_2)$ | $a(s_3,t_3)$
$c(s_1,u_3)$ | $a(s_1,t_3)$	$c(s_2,u_3)$ | $a(s_2,t_3)$	$c(s_3,u_3)$ | $a(s_3,t_3)$
$c(s_1,u_4)$ | $a(s_1,t_3)$	$c(s_2,u_4)$ | $a(s_2,t_3)$	$c(s_3,u_4)$ | $a(s_3,t_3)$
$c(s_1,u_5)$ | $a(s_1,t_3)$	$c(s_2,u_5)$ | $a(s_2,t_3)$	$c(s_3,u_5)$ | $a(s_3,t_3)$
$c(s_1,u_1)$ | $a(s_1,t_4)$	$c(s_2,u_1)$ | $a(s_2,t_4)$	$c(s_3,u_1)$ | $a(s_3,t_4)$
$c(s_1,u_2)$ | $a(s_1,t_4)$	$c(s_2,u_2)$ | $a(s_2,t_4)$	$c(s_3,u_2)$ | $a(s_3,t_4)$
$c(s_1,u_3)$ | $a(s_1,t_4)$	$c(s_2,u_3)$ | $a(s_2,t_4)$	$c(s_3,u_3)$ | $a(s_3,t_4)$
$c(s_1,u_4)$ | $a(s_1,t_4)$	$c(s_2,u_4)$ | $a(s_2,t_4)$	$c(s_3,u_4)$ | $a(s_3,t_4)$
$c(s_1,u_5)$ | $a(s_1,t_4)$	$c(s_2,u_5)$ | $a(s_2,t_4)$	$c(s_3,u_5)$ | $a(s_3,t_4)$

Computing $\mathcal{VA}$. We put the ideas above in a general form by introducing *combinations of sat-pairs* and a function on them leading directly to a va-pair.

Definition 5 ($\mathcal{CSP}$). *A multiset*

$$CSP = \{\!\{(satC_1, satA_1), \ldots, (satC_{G_{\mathrm{C}\cap\mathrm{A}}}, satA_{G_{\mathrm{C}\cap\mathrm{A}}})\}\!\}$$

consisting of $G_{\mathrm{C}\cap\mathrm{A}}$-many sat-pairs $(satC_i, satA_i) \in \mathcal{SP}$ is called a combination of sat-pairs *of r. The set*

$$\mathcal{CSP} := \{CSP \mid CSP \text{ is a combination of sat-pairs}\}$$

is the set of all possible combinations of sat-pairs *of r.*

Definition 6. *The* va-pair function $\mathrm{va}_{\mathcal{C}} : \mathcal{CSP} \to \mathcal{VA}$ on combinations of sat-pairs of r *is defined by* $\mathrm{va}_{\mathcal{C}}(CSP) := \langle v, a \rangle$ *with*

$$v = \sum_{(satC, satA) \in CSP} satC \cdot satA \qquad \text{and} \qquad a = G_{\mathrm{C}\setminus\mathrm{A}} \sum_{(satC, satA) \in CSP} satA$$

In contrast to the function va_{Ω} (Def. 1), the function $\mathrm{va}_{\mathcal{C}}$ only considers combinations of sat-pairs instead of worlds.

Proposition 3. *Let r be an atomic conditional. Then we have*

$$\mathcal{VA} = \mathrm{va}_{\Omega}(\Omega) = \bigcup_{\omega \in \Omega} \{\mathrm{va}_{\Omega}(\omega)\} \tag{11}$$

$$= \mathrm{va}_{\mathcal{C}}(\mathcal{CSP}) = \bigcup_{CSP \in \mathcal{CSP}} \{\mathrm{va}_{\mathcal{C}}(CSP)\} \tag{12}$$

Proposition 3 states that the set $\mathcal{VA}$ can as well be determined via combinations of sat-pairs instead of considering worlds. In particular, (12) gives rise to an algorithm which computes $\mathcal{VA}$ by just running over the set $\mathcal{CSP}$, i. e. without taking into account the much larger set Ω at any point.

The size of $\mathcal{CSP}$ is determined by the well-known multiset coefficient $\left(\!\binom{n}{k}\!\right) = \binom{n+k-1}{k}$, which denotes number of multisets of cardinality k (here: $k = G_{\mathrm{C}\cap\mathrm{A}}$) with elements taken from a set of cardinality n (here: $n = |\mathcal{SP}|$):

Proposition 4. *The number of combinations of sat-pairs with respect to r is:*

$$|\mathcal{CSP}| = \left(\!\binom{|\mathcal{SP}|}{G_{\mathrm{C}\cap\mathrm{A}}}\!\right) = \binom{|\mathcal{SP}|+G_{\mathrm{C}\cap\mathrm{A}}-1}{G_{\mathrm{C}\cap\mathrm{A}}} = \binom{G_{\mathrm{C}\cap\mathrm{A}}+G_{\mathrm{C}\setminus\mathrm{A}}+G_{\mathrm{A}\setminus\mathrm{C}}+G_{\mathrm{C}\setminus\mathrm{A}}\cdot G_{\mathrm{A}\setminus\mathrm{C}}}{G_{\mathrm{C}\cap\mathrm{A}}}$$

Example 4. In Ex. 3, we have $|\Omega| = 2^{|\mathcal{H}|} = 2^{G_{\mathrm{C}\cap\mathrm{A}}\cdot(G_{\mathrm{C}\setminus\mathrm{A}}+G_{\mathrm{A}\setminus\mathrm{C}})} = 2^{27} = 134{,}217{,}728$ worlds compared to just $|\mathcal{CSP}| = \binom{3+5+4+5\cdot 4}{3} = \binom{32}{3} = 4{,}960$ combinations of sat-pairs of $r_{X_s Z_u | X_s Y_t}$. Both sets induce the $|\mathcal{VA}| = 348$ va-pairs of $r_{X_s Z_u | X_s Y_t}$ according to Prop. 3.

We get similar magnitudes of numbers by analyzing other examples involving atomic conditionals (e. g. as in Ex. 2), so we can state in general $|\Omega| \gg |\mathcal{CSP}|$.

Computing Weights of va-Pairs. Up to this point, we have achieved the first part of our goal: determining the WCI $(\mathcal{VA}, \mathrm{wgt})$ of an atomic conditional without any involvement of Ω. So we still have to show how the weight $\mathrm{wgt}(\langle v, a\rangle)$ of each va-pair can be determined also without considering Ω.

According to (8), the weight $\mathrm{wgt}(\langle v, a\rangle)$ of a va-pair corresponds to the number of worlds from Ω which induce $\langle v, a\rangle$. Thus, it seems hard to develop a closed-formed expression which directly provides the weight of a va-pair without considering Ω. However, Prop. 3 showed us that it is feasible to consider combinations of sat-pairs instead of worlds in a certain situation. Furthermore, in the previous examples we illustrated how worlds induce sat-pairs and combinations of sat-pairs, respectively. By employing basic techniques from the field of combinatorics, we indeed obtain a closed formula:

Proposition 5 (nw)**.** *Let $CSP \in \mathcal{CSP}$ be a combination of sat-pairs of r. Let k be number of different sat-pairs contained in CSP and let $m_1, \ldots, m_k$ be the multiplicities of these sat-pairs. Then the function* $\mathrm{nw} : \mathcal{CSP} \to \mathbb{N}$ *with*

$$\mathrm{nw}(CSP) := \binom{G_{\mathrm{C}\cap\mathrm{A}}}{m_1, \ldots, m_k} \cdot \prod_{\substack{(satC, satA) \\ \in CSP}} \binom{G_{\mathrm{C}\setminus\mathrm{A}}}{satC} \cdot \binom{G_{\mathrm{A}\setminus\mathrm{C}}}{satA}$$

yields the number of worlds inducing a combination of sat-pairs.

Input: - a knowledge base $(\Sigma, \{r\})$ with r being an atomic condtional
Output: - the weighted conditional impacts $(\mathcal{VA}, \mathrm{wgt})$

1. $\mathcal{VA} := \emptyset$ *// initialize set*
2. for each $CSP \in \mathcal{CSP}$ *// construct the next CSP systematically on demand*
   (a) $\langle v, a\rangle := \mathrm{va}_{\mathcal{C}}(CSP)$ *// compute the va-pair induced by CSP by eval. the //funct.* $\mathrm{va}_{\mathcal{C}}$
   (b) if $\mathcal{VA} \cap \{\langle v, a\rangle\} = \emptyset$ *// check if* $\langle v, a\rangle$ *appears for the first time*
   then
      i. $\mathcal{VA} := \mathcal{VA} \cup \{\langle v, a\rangle\}$ *// adjoin* $\langle v, a\rangle$ *to the set* $\mathcal{VA}$
      ii. $\mathrm{wgt}(\langle v, a\rangle) := 0$ *// initialize function value*
   (c) *// compute the number of worlds inducing CSP by evaluating the function //*nw *and*
   // increase the weight of $\langle v, a\rangle$ *by* $\mathrm{nw}(CSP)$
   $\mathrm{wgt}(\langle v, a\rangle) := \mathrm{wgt}(\langle v, a\rangle) + \mathrm{nw}(CSP)$
   end loop

Fig. 1. Algorithm $\mathtt{WCI}_{\mathrm{at}}$ yielding $(\mathcal{VA}, \mathrm{wgt})$ of an atomic conditional in time $\mathcal{O}(|\mathcal{CSP}|)$

The above equation makes use of the well-known number of multiset permutations [1], which is defined as $\binom{G_{\mathrm{C}\cap\mathrm{A}}}{m_1,\ldots,m_k} = \frac{G_{\mathrm{C}\cap\mathrm{A}}!}{m_1!\cdot\ldots\cdot m_k!}$. For instance, by applying the function nw to the combination of sat-pairs CSP''' given in (10), where $G_{\mathrm{C}\cap\mathrm{A}} = 3$, we get:

$$\mathrm{nw}(CSP''') = \binom{3}{1,\,2} \cdot \binom{5}{3} \cdot \binom{4}{2} \cdot \binom{5}{3} \cdot \binom{4}{2} \cdot \binom{5}{1} \cdot \binom{4}{4} = 54{,}000.$$

That is, besides the world ω''' from above, there is a total number of 54,000 worlds in Ω which induce CSP'''.

5 Practical Algorithm and Evaluation

Although the function nw in Prop. 5 does not provide the weight of a va-pair directly, it allows us develop an algorithm which successively computes all function values of wgt in parallel without considering Ω.

The algorithm $\mathtt{WCI}_{\mathrm{at}}$ depicted in Fig. 1 takes an atomic conditional r together with an appropriate signature Σ and computes $(\mathcal{VA}, \mathrm{wgt})$ by running over all elements from $\mathcal{CSP}$ once. In step (2), the main loop runs over each multiset $CSP \in \mathcal{CSP}$. Note that the next multiset CSP can be constructed systematically on demand each time, so that the algorithm does not have to store the set $\mathcal{CSP}$ at any point. Within the loop, the function $\mathrm{va}_{\mathcal{C}}$ (see Prop. 3) is employed for the current CSP to determine the corresponding va-pair $\mathrm{va}_{\mathcal{C}}(CSP)$. That way, the set $\mathcal{VA}$ is constructed successively in step (2(b)i). In step (2c), the number of worlds inducing CSP is determined by employing the function nw (see Prop. 5), and $\mathrm{nw}(CSP)$ is added to the current weight of the respective va-pair, i. e. the

Table 6. Results for computing the weighted conditional impacts $(\mathcal{VA}, \mathrm{wgt})$ for different atomic conditionals and numbers of constants; the numbers of constants for the sorted conditional $r_{X_s Z_u | X_s Y_t}$ refer to the sorts S, U, and T, respectively.

Atomic Conditional	$\lvert Const\rvert$	$\lvert\Omega\rvert$	$\lvert\mathcal{CSP}\rvert$	$\lvert\mathcal{VA}\rvert$	Runtime of Algorithm $\texttt{WCI}_{\text{gen}}$	Runtime of Algorithm $\texttt{WCI}_{\text{at}}$
$r_{X\mid X}$	10	2^{20}	286	66	3 sec	<1 sec
	12	2^{24}	455	91	29 sec	<1 sec
	15	2^{30}	816	136	30 min 32 sec	<1 sec
	50	2^{100}	23,426	1,326	*unfeasible*	<1 sec
$r_{X\mid XY}$	4	2^{20}	715	129	2 sec	<1 sec
	5	2^{30}	4,368	301	56 min 38 sec	<1 sec
	10	2^{100}	44,352,165	4,701	*unfeasible*	1 min 13 sec
	12	2^{156}	1,852,482,996	9,793	*unfeasible*	1 h 2 min
$r_{X_s Z_u \mid X_s Y_t}$	3+5+4	2^{27}	4,960	348	13 min 46 sec	<1 sec
	3+6+5	2^{33}	13,244	644	≈ 27 hours	<1 sec
	3+10+10	2^{60}	302,621	4,215	*unfeasible*	<1 sec
	4+10+10	2^{80}	9,381,251	7,685	*unfeasible*	11 sec
	5+10+10	2^{100}	234,531,275	12,155	*unfeasible*	5 min 22 sec
	6+10+10	2^{120}	4,925,156,775	17,625	*unfeasible*	1 h 53 min

va-pair which is induced by *CSP*. That way, the correct weight $\mathrm{wgt}(\langle v, a\rangle)$ of each va-pair is computed incrementally.

Note that the set Ω is not considered in any step of the algorithm $\texttt{WCI}_{\text{at}}$. In particular, the functions $\mathrm{va}_{\mathcal{C}}$ and nw employed by the algorithm are closed-form expressions which do not refer to Ω either. Thus, the overall runtime of the algorithm is determined by the size of the set $\mathcal{CSP}$, and the algorithm merely requires space $\mathcal{O}(|\mathcal{VA}|)$, which is no more than the size of the output. These observations yield:

Proposition 6. *The algorithm* $\texttt{WCI}_{\text{at}}$ *in Fig. 1 computes the WCI* $(\mathcal{VA}, \mathrm{wgt})$ *of an atomic conditional in time* $\mathcal{O}(|\mathcal{CSP}|)$ *and space* $\mathcal{O}(|\mathcal{VA}|)$.

Table 6 shows some results for applying algorithm $\texttt{WCI}_{\text{at}}$ to some conditionals from Ex. 2 and 3 together with various numbers of constants. For instance, for $r_{X_s Z_u | X_s Y_t}$ together with a set of 3, 10, and 10 constants of the respective sorts, the generic algorithm $\texttt{WCI}_{\text{gen}}$ from [5], requiring time $\mathcal{O}(|\Omega|)$, must process $2^{60} \approx 10^{18}$ worlds which is already infeasible, whereas the algorithm $\texttt{WCI}_{\text{at}}$ merely has to consider $|\mathcal{CSP}| = 302{,}621$ combinations of sat-pairs, taking less than one second to determine the weighted conditional impacts $(\mathcal{VA}, \mathrm{wgt})$. Thus, $\texttt{WCI}_{\text{at}}$ allows to determine the actual set $\mathcal{VA}$ and its actual size in particular for an increasing number of constants; due to the obvious limitations of any generic algorithm working on Ω, such concrete numbers could not be computed before. The actual numbers for $|\mathcal{VA}|$ in Tab. 6 suggest that the number of va-pairs grows significantly slower than $|\mathcal{CSP}|$, i. e. we have:

$$|\Omega| \gg |\mathcal{CSP}| \gg |\mathcal{VA}|$$

Indeed, the size of $\mathcal{CSP}$, given by the binomial coefficient in Prop. 4, also grows rather fast in the number of constants. Nevertheless, it grows significantly slower than the size of Ω, which grows exponentially in a polynomial of the constants. So our novel algorithm $\texttt{WCI}_{\text{at}}$ also serves as a proof of concept and illustrates that it is in fact possible to determine $(\mathcal{VA}, \text{wgt})$ without considering the set Ω at all.

6 Conclusions and Future Work

The WCI of a relational probabilistic knowledge base are the essential ingredient for Maximum Entropy model computation. We presented a new approach allowing to compute the WCI of an atomic conditional based on combinatorics. The resulting algorithm fully abstracts from Ω, using the set $\mathcal{CSP}$ instead, and since $|\Omega| \gg |\mathcal{CSP}|$, it is much faster than a generic algorithm which has to run over Ω. In terms of a first proof of concept, we restricted our considerations to a single atomic conditional in this paper. In future work, we will investigate how the general concepts introduced here can be extended to more complex conditionals and to more than one conditional. For instance, a non-atomic conditional $(\, c(X) \mid a(X) \wedge b(Y)\,)$ can be transformed into an atomic conditional $(\, c(X) \mid aAndb(X,Y)\,)$ by introducing a new predicate $aAndb(X,Y)$ which captures the non-atomic formula in the antecedence. Furthermore, while the development of an incremental algorithm by extending the WCI appropriately when adding another conditional may not be feasible in all cases, the current concept can directly be extended to, e. g. a set of several atomic conditionals if we ensure that each conditional considers different predicates. We will also investigate to what extent techniques of lifted inference [3,9,16] can be adopted to our approach.

References

1. Brualdi, R.A.: Introductory Combinatorics (5th Edition). Pearson (2009)
2. Darroch, J.N., Ratcliff, D.: Generalized iterative scaling for log-linear models. Annals of Mathematical Statistics **43**(5), 1470–1480 (1972)
3. de Salvo Braz, R., Amir, E., Roth, D.: Lifted first-order probabilistic inference. In: IJCAI-05, pp. 1319–1325. Professional Book Center (2005)
4. Finthammer, M., Beierle, C.: Using equivalences of worlds for aggregation semantics of relational conditionals. In: Glimm, B., Krüger, A. (eds.) KI 2012. LNCS, vol. 7526, pp. 49–60. Springer, Heidelberg (2012)
5. Finthammer, M., Beierle, C.: A two-level approach to maximum entropy model computation for relational probabilistic logic based on weighted conditional impacts. In: Straccia, U., Calì, A. (eds.) SUM 2014. LNCS, vol. 8720, pp. 162–175. Springer, Heidelberg (2014)
6. Fischer, V., Schramm, M.: Tabl - a tool for efficient compilation of probabilistic constraints. Technical Report TUM-19636, Technische Universität München (1996)
7. Getoor, L., Taskar, B. (eds.): Introduction to Statistical Relational Learning. MIT Press (2007)

8. Halpern, J.: Reasoning About Uncertainty. MIT Press (2005)
9. Kazemi, S.M., Buchman, D., Kersting, K., Natarajan, S., Poole, D.: Relational logistic regression. In: Proc. 14th International Conference on Principles of Knowledge Representation and Reasoning (KR-2014) (2014)
10. Kern-Isberner, G.: Conditionals in Nonmonotonic Reasoning and Belief Revision. LNCS, vol. 2087. Springer, Heidelberg (2001)
11. Kern-Isberner, G., Beierle, C., Finthammer, M., Thimm, M.: Comparing and evaluating approaches to probabilistic reasoning: Theory, implementation, and applications. Transactions on Large-Scale Data- and Knowledge-Centered Systems **6**, 31–75 (2012)
12. Kern-Isberner, G., Lukasiewicz, T.: Combining probabilistic logic programming with the power of maximum entropy. Artif. Intell. **157**(1–2), 139–202 (2004)
13. Kern-Isberner, G., Thimm, M.: Novel semantical approaches to relational probabilistic conditionals. In: Proc. KR-2010, pp. 382–391. AAAI Press, Menlo Park, CA (2010)
14. Kern-Isberner, G., Thimm, M.: A ranking semantics for first-order conditionals. In: ECAI-2012, pp. 456–461. IOS Press (2012)
15. Lukasiewicz, T.: Probabilistic logic programming with conditional constraints. ACM Trans. Comput. Logic **2**(3), 289–339 (2001)
16. Milch, B., Zettlemoyer, L., Kersting, K., Haimes, M., Kaelbling, L.P.: Lifted probabilistic inference with counting formulas. In: AAAI-2008, pp. 1062–1068. AAAI Press (2008)
17. Nilsson, N.: Probabilistic logic. Artificial Intelligence **28**, 71–87 (1986)
18. Paris, J.: The uncertain reasoner's companion - A mathematical perspective. University Press, Cambridge (1994)
19. Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann, San Mateo, Ca (1988)
20. Poole, D.: First-order probabilistic inference. In: Gottlob, G., Walsh, T. (ed.), Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03), pp. 985–991. Morgan Kaufmann (2003)
21. Potyka, N., Beierle, C., Kern-Isberner, G.: A concept for the evolution of relational probabilistic belief states and the computation of their changes under optimum entropy semantics. Journal of Applied Logic (2015). (to appear)
22. Richardson, M., Domingos, P.: Markov logic networks. Machine Learning **62**(1–2), 107–136 (2006)
23. Shore, J., Johnson, R.: Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. IEEE Transactions on Information Theory, IT-26:26–37 (1980)