Reasoning with generics and induction*

Liying Zhang
The Institute of Modern Logic, Central University of
Finance and Economics Beijing, 100081
clearliying@126.com

Abstract

Based on the results of reasoning with generics, this paper attempts to clarify the relationship between generic reasoning and inductive reasoning. First, to capture reasoning with generics, logics for obtaining intermediate conclusions are provided; then, the priority order of the subsets of the premise set are introduced to eliminate the contradictions or incompatibility generated by intermediate conclusions. Through these filters, the final conclusions are obtained. There are different priority rules on subsets of a premise set corresponding to three kinds of generic reasoning: reasoning with factual sentences (from generic sentences), reasoning by deduction and reasoning by induction. If these three kinds of reasoning are combined, together they can characterise inductive reasoning. After reviewing how scholars' have previously interpreted inductive reasoning, this paper concludes that there is another way to interpret inductive reasoning with generic reasoning, parallel to the probability method.

1 Reasoning with generics

Generic sentences, for instance, '*birds fly*' or '*ducks lay eggs*', express rules or laws. Unlike universal sentences, generic sentences tolerate exceptions. Even when there is no positive example in the real world (for example, '*unicorns have one horn*'), we sometimes accept a generic sentence, which makes generic sentences intensional. Because reasoning with generics tolerates exceptions, it is non-monotonic.

1.1 Interpretation of generic sentences

This paper restricts its scope to i-generics[1] and employs the semantics from Mao and Zhou (2003) for generics: The canonical form of a generic sentence with subject-predicate (SP) structure is '(normal S) (normally P)'. The duty of 'normal' is to choose a set of (normal) objects (the term's extension) for every possible world based on both a subject sense and predicate sense. 'Normally' is used to choose the normal situation.

The sense of the term 'S' can be expressed as $\lambda x S$ by λ-expression. From a semantic aspect, a sense is a function expressed as s, and 'normal' is a binary function $\mathcal{N}(s_1, s_2)$ called the normal function. s_1, s_2 respectively

* This work was supported by "the Fundamental Research Funds for the Central Universities".

In: T. Ågotnes, B. Liao, Y.N. Wang (eds.): Proceedings of the first Chinese Conference on Logic and Argumentation (CLAR 2016), Hangzhou, China, 2-3 April 2016, published at http://ceur-ws.org

[1]Krifka et al. (1995) distinguish between d-generic and i-generic sentences. '*Dinosaurs are extinct*' is an example of d-generic sentence, since a type of animal can be extinct but individual members of that type can only be dead. '*Birds fly*' is a famous example of an i-generic sentence.

represent the subject sense and the predicate sense. This means that the normal subject's sense is ascertained through the subject sense and the predicate sense. There are two basic restrictive conditions on s. First, for any sense $s_1, s_2, \mathcal{N}(s_1, s_2) \subseteq s_1$; i.e., the normal subject sense (chosen by $\mathcal{N}$) is included in the subject sense. Second, for any sense $s_1, s_2, \mathcal{N}(s_1, s_2) = \mathcal{N}(s_1, \tilde{s_2})$; i.e., the normal subject sense has a relationship with the predicate sense regardless of whether the predicate is affirmative or negative. Due to these two restrictions, the drowning problem[2] can be solved, and a cyclical definition caused by choosing a normal subject based on a predicate can be avoided.

'Normally' is represented by '>', which comes from conditional logic used by Pelletier and Asher (1997), improved by Mao and Zhou (2003). Based on possible worlds semantics, the intension of a generic sentence can be expressed, and a sentence like '*unicorns have one horn*' can be interpreted.

Based on the points above, '(normal S) (normally P)' can be interpreted further: for any object x, if x is normal S regarding P or not P, then, normally, x is P. According to this analysis, a generic sentence SP's formalization is $\forall x(\mathcal{N}(\lambda x Sx, \lambda x Px)x > Px)$, and $G(Sx; Px)$.[3] For short, G is called a generic quantifier.

There is also an important interpretation of generics by Cohen (1999) using probability techniques; however, that method through probability techniques is doomed to be inadequate for interpreting intensional generics.

To ease readers and make the structure of this paper clearer, in section 1.1, 1.2, 1.3, I only give the main ideas and main conclusions. The technique details are left in section 1.4.

1.2 Reasoning with generics

When reasoning with generics, there are two questions: what can be obtained from generic sentences and how are generic sentences obtained? These two parts are mixed since from generic sentences and factual sentences we can obtain both generic sentences and factual sentences. For historical and technical reasons, in this paper reasoning with generics is divided into two different parts (not a partition): conclusions are factual propositions or conclusions are generic sentences. The former reasons that generic sentences are common sense, as they reach conclusions about concrete facts under concrete circumstances. For instance, from '*birds fly*' and '*Tweety is a bird*', the conclusion is '*Tweety flies*'. The latter reasons from some propositions and reaches general conclusions represented by generic sentences. This kind of reasoning involves both deduction (in which premises may contain generic sentences) and induction (in which premises may not contain generic sentences). At first glance, induction seems more important and interesting; however, in daily life, reasoning by deduction with generics is more common. For instance, from '*sparrows are birds*' and '*birds have feathers*' we get '*sparrows have feathers*', one of the thousands of inferences leading to a generic sentence by deduction. Because there is an enormous amount of knowledge and belief in our brains represented by generic sentences, reasoning with pure induction in our daily lives approaches an ideal.

Based on the semantics provided in section 1.1., there are different logics corresponding to the different parts of generic reasoning. Different from classical logic where the logics can be used to capture the deductive reasoning, in the paper, the logics captured partial reasoning with generics; the logics within the priority ordering on the premise set will capture the whole reasoning process. This means that logics are used to capture the inferences contained in each reasoning step. When these inferences' conclusions are combined, some partial conclusions may lead to disorder and even conflict. Then, the priority ordering of premise sets work, drawing back some 'conclusions' and leaving the final conclusions.

1.2.1 Reasoning with generics when conclusions are factual propositions

A famous example of reasoning with generics in which the conclusions are factual propositions is '*birds fly*' and '*Tweety is a bird*', therefore '*Tweety flies*'. In Mao (2003), logics M and G are constructed to address inferences that reach factual conclusions from general premises. In M, the default implication MP inference is considered: $(\alpha \wedge (\alpha > \beta)) > \beta$. Beginning with this formula, G, as the quantifier extension has an important new axiom GU: $Gx(\alpha; \beta) > \forall x(\alpha > \beta)$. Its intuition is: normally, if '(normal S) (normally P)', then 'S normally P'. With GU, Mao (2003) transforms the generic quantifier into a universal quantifier. In the substitution example, the universal quantifier can be eliminated, and from the default MP rule implication, inferred from the generic proposition and other specific conditions, we reach the factual conclusion accordingly.

[2]Both '*peacocks lay eggs*' and '*peacocks have colorful feathers*' are true generic sentences, but only female peacocks lay eggs and only male peacocks have colorful feathers.

[3]Sometimes it is written as $Gx(\alpha; \beta)$.

1.2.2 Obtaining generic sentences mainly by deduction

Based on the similar interpretation of generics, there is a logic G_D for obtaining generic sentences by deduction. Here, the goal is for the inferences to lead to generic sentences, so we don't need to consider the transformation from generic quantifier to universal quantifier. In the formal system, we don't add the axiom GU; instead, the relationship between generic sentences is considered by studying the normal function $\mathcal{N}$. We will expand and enrich $\mathcal{N}$, which has been discussed above, and obtain some inference rules about generic sentences to obtain, for example, the inference for GAG.

There is an important axiom in G_D:

$$N_{AM} \qquad \forall y(\alpha \to \gamma)(y/x) \to \forall y(N(\lambda x\alpha, \lambda x\beta)y \to N(\lambda x\gamma, \lambda x\beta)y)$$

The corresponding semantic condition (subject-monotonic) is:

For any $s_1, s_2, s_3 \in \mathscr{P}(D)^W$, any $w \in W$, if $s_1(w) \subseteq s_2(w)$, then $\mathcal{N}(s_1, s_3)(w) \subseteq \mathcal{N}(s_2, s_3)(w)$.

The intuition is that, if in the same possible world w the object sets corresponding to s_1 are included in the object sets corresponding to s_2, then for the same predicate sense s_3, in w, the object sets according to normal s_1 are included in the object sets corresponding to s_2. For instance, in the actual world, there is: '*sparrows are birds*', '*birds fly*', '*sparrows fly*'. Corresponding to 'fly' in the actual world, 'normal sparrow(s)' are also normal bird(s)'.

1.2.3 Obtaining generic sentences mainly by induction

If there is no generic sentence in the premise set then to obtain a generic sentence the only option is simple enumeration. Does that sound too simple? That is it. If there is something more complex, then the premise set must contain higher-order knowledge or belief, i.e., generic sentences, then the inferences come down to the scope of G_D.

1.3 Priority orders on premise set

There are logics for different parts of generic reasoning. However, this is not enough, for example, based on the system $\mathrm{G_D}$. From {*sparrows are birds, birds fly*} we get *sparrows fly*; also, from {*penguins are birds, birds fly*}, we get *penguins fly.* But what we can get from {*penguins are birds, birds fly, penguins don't fly*}? Introducing priority orders on a premise set is a feasible and reasonable way to deal with the potential disorder.

The aim is to create an order for the subsets of the premise set. The order among subsets can be defined by the order among the formulas in the premise set. So, we first define the strict partial order among the formulas in the premise set. Then, we present normal rules for priority order and special rules for different kinds of reasoning. As an example, details of the orders on G_D are in section 1.4. Next, I will present some special orders for different parts of reasoning.

1.3.1 Special priority orders for reasoning obtaining factual propositions

Zhou and Mao (2004) have placed priority order on the premise set and shown that their definition can pass through benchmark examples such as Nixon Diamond, Penguin Principle, etc.

(1) Factual priority: if $\alpha > \beta, \neg\beta \in \Gamma$, then $\neg\beta \succ_G \alpha > \beta$,

(2) Sub-category default priority: if $\alpha > \beta \in \Gamma$, and $\alpha > \gamma, \beta > \delta \in \Gamma$, then $\alpha > \gamma \succ_G \beta > \delta$.

1.3.2 Special priority orders for reasoning obtaining generics by deduction

Moving from '*birds fly*' and '*sparrows are birds*' to '*sparrows fly*' (GAG type) is a representative instance of reasoning obtaining generics by deduction. There are two special rules for this kind of reasoning: sub-category generics priority and generics priority. Our definition can plainly pass through the GAG.

(1) For any formula $\alpha, \beta, \gamma, \delta$, if $\forall x(\alpha \to \beta) \in \Gamma$ and $Gx(\alpha;\gamma), Gx(\beta;\delta) \in \Gamma$, then $Gx(\alpha;\gamma) \succ_G Gx(\beta;\delta)$ (sub-category generics priority);

(2) For any formula α, β, if $\forall x(\alpha \to \beta) \in \Gamma$, and $Gx(\alpha;\gamma) \in \Gamma$, then $Gx(\alpha;\gamma) \succ_G \forall x(\alpha \to \beta)$. (generics priority)

1.3.3 Special priority orders for reasoning obtaining generics by induction

Now we reach the most difficult one: obtaining generic sentences by induction. As I say in 1.2.3., simple enumeration is the way to obtain generics by induction. But why, even there are many examples of white swans, do we not accept '*swans are white*' to be a true generic sentence by simple enumeration? One explanation is that the premise contains a generic sentence (or the propositions to reach it) such as '*poultry feathers are multiple colours*'. This can be obtained by simple enumeration from '*ducks' feathers are multiple colours'*, '*parrots' feathers are multiple colours*', '*domestic gooses' feathers are multiple colours*'. But why we accept this conclusion rather than '*swans are white*'? They all come from simple enumeration. The answer is again the ordering of the premise sets. Here, for inferences obtaining generic sentences by induction, the priority order is *category generics priority.*[4]

The special priority order for obtaining generic sentences by induction is:

For any formula $\alpha, \beta, \gamma, \delta$, if $\forall x(\alpha \to \beta) \in \Gamma$ and $Gx(\alpha;\gamma), Gx(\beta;\delta) \in \Gamma$, then $Gx(\beta;\delta) \succ_G Gx(\alpha;\gamma)$ (category generics priority).

This has provided an overview of the work on reasoning with generics. Below, we begin to discuss induction and the relationship between it and reasoning with generics.

1.4 Formalization

1.4.1 Language and semantics

Formal language $\mathscr{L}_G$ contains a denumerable set of individual variables (Var), a denumerable set of individual constants and a denumerable set of predicate variables ($n > 0$), a sentential constant $\bot$, a truth-functional operator $\to$, universal quantifier $\forall$, parenthesis), (. These are the symbols of first-order language. Based on them, we add new symbols: $>, \lambda, N$.

We use x, y, z, etc., for any individual variable, c, d, e, etc., for any individual constant, P, Q etc., for any predicate, t, etc., for any term, $\alpha, \beta, \gamma, \varphi$, etc., for any formula.

$$\alpha ::= \bot | Pt | \alpha \to \beta | \forall x\ \alpha | \alpha > \beta | (\lambda x \alpha)t | N(\lambda x \alpha, \lambda x \beta)t |$$

We introduce symbols in $\mathscr{L}_G$ such as $\top, \neg, \wedge, \vee, \leftrightarrow, \exists$, following the usual definitions.

$$Gx(\alpha;\beta) =_{\text{df}} \forall x(N(\lambda x\alpha, \lambda x\beta)x > \beta)$$

Definition 1. *W is a non-empty set. Function $\circledast$: $\mathscr{P}(W) \times \mathscr{P}(W) \to \mathscr{P}(W)$ is a set selection function on W, if for any $X, Y, Z, X', Y' \subseteq W, \circledast$ satisfies:*

(1) If $X \subseteq X'$, then $\circledast(X,Y) \subseteq \circledast(X',Y)$.

(2) If $\circledast(\{w\},Y) \subseteq Z$ for every $w \in X$, then $\circledast(X,Y) \subseteq Z$.

(3) If $\circledast(X,Y) \subseteq Z$, then $\circledast(W, X \cap Y) \subseteq Z$.

(4) $\circledast(X,Y) \subseteq Y$.

(5) If $\circledast(X,Y) \subseteq Z$ and $\circledast(X,Y') \subseteq Z$, then $\circledast(X, Y \cup Y') \subseteq Z$.

(6) If $\circledast(X,Y) \subseteq Y'$ and $\circledast(X,Y') \subseteq Z$, then $\circledast(X,Y) \subseteq Z$.

Definition 2. *W and D are non-empty sets. $S = \mathscr{P}(D)^W$, For any $s_1, s_2 \in S$,*

(1) $s_1 \subseteq s_2$ iff. $s_1(w) \subseteq s_2(w)$ for all $w \in W$.

(2) $s_1 = s_2$ iff. $s_1(w) \subseteq s_2(w)$ and $s_2(w) \subseteq s_1(w)$ for all $w \in W$.

(3) $s_1 = s_2^{\sim}$ iff. $s_1(w) = (s_2(w))^{\sim}$ for all $w \in W$. ($A^{\sim}$ is the supplementary set of A.)

(4) For each $w \in W, (s_1 \cup s_2)(w) = s_1(w) \cup s_2(w), (s_1 \cap s_2)(w) = s_1(w) \cap s_2(w)$.

(5) Vacant sense $s_\top, s_\top(w)$=D for all $w \in W$.

[4]In fact, the generics priority in 1.3.2., above, also works through this process. There are more details in Zhang (2009).

(6) Full sense $s_\perp, s_\perp(w) = \varnothing$ for all $w \in W$.

Definition 3. *W and D are non-empty sets, $S = \mathscr{P}(D)^W$. $\mathcal{N} : S \times S \to S$ is a normal object selection function .iff. For all $s_1, s_2 \in S, \mathcal{N}$ satisfies:*

(1) $\mathcal{N}(s_1, s_2) \subseteq s_1$, and

(2) $\mathcal{N}(s_1, s_2) = \mathcal{N}(s_1, s_2^{\sim})$.

Definition 4. *W and D are non-empty sets, $S = \mathscr{P}(D)^W$. $[,] : D \times S \to \mathscr{P}(W)$ is a function that satisfies: for all $d \in D, s \in S, [d, s] = \{w \in W : d \in s(w)\}$.*

Definition 5. *Quadruple $F = <W, D, \mathcal{N}, \circledast>$ is a frame; if W and D are non-empty sets, then $\mathcal{N}$ is a normal object selection function on S and $\circledast$ is a set selection function on W.*

Definition 6. *$F = <W, D, \mathcal{N}, \circledast>$ is a frame η is interpretation on F, .iff.*

(1) For each individual constant $c \in C, \eta(c) \in D$.

(2) For each predicate symbol $P, \eta(P, w) \in D$.

Definition 7. *An ordered pair $S = <F, \eta>$ is a structure if F is a frame and η is a interpretation function on F.*

Definition 8. *An ordered pair $M = <S, \sigma>$ is a model; if S is a structure σ is a mapping function from Var to D. We also call σ an assignment.*

We use F_M, S_M to denote the frame and structure of M, using $W_M, D_M, \eta_M, \sigma_M$, etc., to spell out in detail components of the model M. Given term t, the interpretation of t in M is denoted by t^M (the usual definition).

Definition 9. *For each formula φ, the symbol $\|\varphi\|^M$ is used to stand for the set of worlds in M in which φ is true, satisfying:*

(1) $\|\perp\|^M = \emptyset$

(2) $\|Pt\|^M = \{w \in W : <t^M> \in \eta_M(P, w)\}$

(3) $\|\alpha \to \beta\|^M = (W - \|\alpha\|^M) \cup \|\beta\|^M$

(4) $\|\alpha > \beta\|^M = \cup\{X \subseteq W : \circledast(X, \|\alpha\|^M) \subseteq \|\beta\|^M\}$

(5) $\|\forall x\alpha\|^M = \{w \in W$: for each $d \in D_M, w \in \|\alpha\|^{M(d/x)}\}y$

(6) $\|N(\lambda x\alpha, \lambda x\beta)t\|^M = \{w \in W : t^M \in N((\lambda x\alpha)^M, (\lambda x\beta)^M)(w)\}((\lambda x\alpha)^M \in S$, is a mapping satisfying, for each $w \in W, (\lambda x\alpha)^M(w) = \{d \in D_M : w \in \|\alpha\|^{M(d/x)}\})$.

Proposition 1. *For any variable x, any formulae α, β, any model M:*

(1) $(\lambda x\neg\alpha)^M = ((\lambda x\alpha)^M)^{\sim}$

(2) $(\lambda x(\alpha \vee \beta))^M = (\lambda x\alpha)^M \cup (\lambda x\beta)^M$

(3) $(\lambda x(\alpha \wedge \beta))^M = (\lambda x\alpha)^M \cap (\lambda x\beta)^M$

(4) $(\lambda x(\alpha \to \beta))^M = ((\lambda x\alpha)^M)^{\sim} \cup (\lambda x\beta)^M$

Proposition 2. *For any $M, \|\alpha(y/x)\|^{M(d/y)} = [d^M, (\lambda x\alpha)^M]$*

Proposition 3. *For any model M, any $w \in W, w \in \|\forall y(\beta \to \gamma)(y/x)\|^M$ .iff. $(\lambda x\beta)^M(w) \subseteq (\lambda x\gamma)^M(w)$.*

Corollary 1. *For any model M, any $w \in W, w \in \|\forall y(\beta \leftrightarrow \gamma)(y/x)\|^M$ .iff. $(\lambda x\beta)^M(w) = (\lambda x\gamma)^M(w)$.*

Definition 10. *Let M be a model and α be a formula, $X \subseteq W_M, X \neq \emptyset$. α is true at the set X (written as $M \models_X \alpha$) iff. $X \subseteq \|\alpha\|^M$. When $X = \{w\}$, we also say that α is true at w. When $X = W_M$, we say that α is valid at M; we use $M \models \alpha$ to denote it.*

Definition 11. *Given a formula α, we say that α is valid (written as $\models \alpha$) iff. $M \models \alpha$ for all M.*

Theorem 1. *The formulas listed below are valid.*

(1) $(\forall x(\alpha > \beta) \to (\alpha > \forall x\beta))$ *(x is not a free variable in α)*

(2) $\forall y(N(\lambda x\alpha, \lambda x\beta)y \to \alpha(y/x))$

(3) $\forall y(N(\lambda x\alpha, \lambda x\beta)y \to N(\lambda x\alpha, \lambda x\neg\beta)y)$

(4) $\forall x(\alpha \to \beta) \to \forall x(\alpha > \beta)$ ■

Theorem 2. *(1) If $M \models \beta \leftrightarrow \gamma$ for all model M, then, $M \models \forall y(N(\lambda x\alpha, \lambda x\beta)y \leftrightarrow \forall y(N(\lambda x\alpha, \lambda x\gamma)y)$ for all model M.*

(2) If $M \models \beta \leftrightarrow \gamma$ for all model M, then, $M \models \forall y(N(\lambda x\beta, \lambda x\alpha)y \leftrightarrow N(\lambda x\gamma, \lambda x\alpha)y)$ for all model M. ■

Definition 12. *A frame $<W, D, \mathcal{N}, \circledast>$ is a subject-monotonic frame iff. It satisfies: for any $s_1, s_2, s_3 \in S$, if $s_1(w) \subseteq s_2(w)$, then $\mathcal{N}(s_1, s_3)(w) \subseteq \mathcal{N}(s_2, s_3)(w)$. A model $<W, D, \mathcal{N}, \circledast, \eta, \sigma>$ is a subject-monotonic model iff. $<W, D, \mathcal{N}, \circledast>$ is a subject-monotonic frame.*

Theorem 3. *For any subject monotonic model $M, M \models \forall y(\alpha \to \gamma)(y/x) \to \forall y(N(\lambda x\alpha, \lambda x\beta)y \to N(\lambda x\gamma, \lambda x\beta)y))$.*

1.4.2 Logic G_D for partial inference

Axiom schemata:

T	all tautologies
$\forall^-$	$\forall x\alpha \to \alpha(x/t)$
$\forall_\to$	$\forall x(\alpha \to \beta) \to (\forall x\alpha \to \forall x\beta)$
$>_{\mathrm{BF}}$	$\forall x(\alpha > \beta) \to (\alpha > \forall x\beta)$ (x is not a free variable in α)
C_K	$(\alpha > (\beta \to \gamma)) \to ((\alpha > \beta) \to (\alpha > \gamma))$
$>_{\mathrm{MP}}$	$(\alpha \wedge (\alpha > \beta)) > \beta$
T_{RAN}	$(\alpha > \beta) \to ((\beta > \gamma) \to (\alpha > \gamma))$
A_D	$(\alpha > \gamma) \wedge (\beta > \gamma) \to (\alpha \vee \beta > \gamma)$
I_C	$\forall x(\alpha \to \beta) \to \forall x(\alpha > \beta)$
N	$\forall y(N(\lambda x\alpha, \lambda x\beta)y \to \alpha(y/x))$
$N\neg$	$\forall y(N(\lambda x\alpha, \lambda x\beta)y \to N(\lambda x\alpha, \lambda x\neg\beta)y)$
N_{AM}	$\forall y(\alpha \to \gamma)(y/x) \to \forall y(N(\lambda x\alpha, \lambda x\beta)y \to N(\lambda x\gamma, \lambda x\beta)y))$

Rules of inference:

MP;	$\forall^+$;
R_{CEA}	From $\beta \leftrightarrow \gamma$, infer $(\beta > \alpha) \leftrightarrow (\gamma > \alpha)$;
R_N	From β, infer $\alpha > \beta$;
R_M	From $\alpha > \beta$, infer $(\alpha \wedge \gamma) > \beta$;
R_{NEA}	From $\beta \leftrightarrow \gamma$, infer $\forall y(N(\lambda x\alpha, \lambda x\beta)y \leftrightarrow N(\lambda x\alpha, \lambda x\gamma)y)$;
R_{NEC}	From $\beta \leftrightarrow \gamma$, infer $\forall y(N(\lambda x\beta, \lambda x\alpha)y \leftrightarrow N(\lambda x\gamma, \lambda x\alpha)y)$.

Some theorems and derived rules:

I_D	$\alpha > \alpha$
A_M	$\forall x(\alpha \to \beta) \to \forall x((\beta > \gamma) \to (\alpha > \gamma))$
C_I	$\forall x(\alpha > \beta) \wedge \forall x(\beta \to \gamma) \to \forall x(\alpha > \gamma)$
C_M	$(\alpha > \beta) \to ((\alpha \wedge \gamma) > \beta)$
C_R	$(\alpha > \beta \wedge \gamma) \to ((\alpha > \beta) \wedge (\alpha > \gamma))$
C_C	$((\alpha > \beta) \wedge (\alpha > \gamma)) \to (\alpha > \beta \wedge \gamma)$
$\mathrm{Th_{MN}}1$	$(\alpha > \bot) \to (\alpha > \beta)$
R_N	From β, infer $\alpha > \beta$;
R_{CEA}	From $\beta \leftrightarrow \gamma$, infer $(\beta > \alpha) \leftrightarrow (\gamma > \alpha)$;
R_{CEC}	From $\beta \leftrightarrow \gamma$, infer $(\alpha > \beta) \leftrightarrow (\alpha > \gamma)$;
R_{IC}	From $\alpha \to (\beta > \gamma)$, infer $(\alpha \wedge \beta > \gamma)$;
R_{CK}	From $(\beta_1 \wedge \ldots \wedge \beta_n) \to \beta$, infer $(\alpha > \beta_1) \wedge \ldots \wedge (\alpha > \beta_n) \to (\alpha > \beta); (n \geq 1)$;

R_{CI} From $\alpha > \beta, \beta \to \gamma$, infer $\alpha > \gamma$;
R_{EQ} β is a sub-formula of α, from $\beta \leftrightarrow \gamma$ and α infer $\alpha[\gamma/\beta]$.

Some theorems and derived rules about generic sentences:

$Th_{G_D}1$ $Gx(\alpha;\alpha)$
$Th_{G_D}2$ $Gx(\alpha \wedge \beta;\alpha)$
$Th_{G_D}3$ $\forall y(N(\lambda x\alpha, \lambda x\beta)y \leftrightarrow (N(\lambda x\alpha, \lambda x\neg\beta)y)$
$Th_{G_D}4$ $\forall x(\alpha > \beta) \to Gx(\alpha;\beta)$
$Th_{G_D}5$ $\forall x(\alpha \to \beta) \to Gx(\alpha;\beta)$
$Th_{G_D}6$ $\forall x(\alpha \to \beta) \to (Gx(\beta;\gamma) \to Gx(\alpha;\gamma))$
$Th_{G_D}7$ $Gx(\alpha;\beta) \to Gx(\alpha \wedge \gamma;\beta)$
$Th_{G_D}8$ $Gx(\alpha \vee \beta;\gamma) \to Gx(\alpha;\gamma)$
R_{GN} From β, infer $Gx(\alpha;\beta)$.
R_{GIC} From $\alpha \to Gx(\beta;\gamma)$, infer $Gx(\alpha \wedge \beta;\gamma)$.

By theorems 1, 2, and 3, the system G_D is sound. The completeness of G_D can be proved by a variant of the canonical model method, the details of which can be found in Zhang (2005).

1.4.3 Logic G_D for partial inference

1.4.3.1. The general priority order

Definition 13. *Γ is any formula set. $\succ$ is a (general)priority order, iff., $\succ$ is a 2-ary relation on Γ that satisfies (strictly partial order): for any formula $\alpha, \beta, \gamma \in \Gamma$,*

(1) $\alpha \not\succ \alpha$ (not $\alpha \succ \alpha$);

(2) if $\alpha \succ \beta$ and $\beta \succ \gamma$, then $\alpha \succ \gamma$.

Reasonability

Definition 14. *Let Φ is any formula set. Φ is reasonable, iff., the following two conditions do not occur:*

(1) There is a formula $\alpha, \alpha \in \Phi$ and $\neg\alpha \in \Phi$;

(2) There is a formula $Gx(\alpha;\beta), Gx(\alpha;\beta) \in \Phi$ and $Gx(\alpha;\neg\beta) \in \Phi$.

Definition 15. *Let $\Delta = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$ is any finite formula set.*

$$\wedge\Delta = \begin{cases} \alpha_1 \wedge \alpha_2 \wedge \dots \wedge \alpha_n, & n \geq 1; \\ \bot, & \textit{if not.} \end{cases}$$

Definition 16. *Let Δ is any finite formula set, $S(L)$ is a formal system of logic L.*

(1) $Cn(\Delta)$ is L-consequence set of Δ, if $Cn(\Delta) = \{\alpha : \vdash S(L) \wedge \Delta \to \alpha\}$;

(2) $CN(\Delta)$ is reasonable L-consequence set of Δ, if

$$CN(\Delta) = \begin{cases} Cn(\Delta), & \textit{if } Cn(\Delta) \textit{ is reasonable}, \\ \emptyset, & \textit{if not.} \end{cases}$$

(3) For any $\alpha \in CN(\Delta), \Delta$ is L-premise set of α. If for any Δ's proper subset $\Delta', \alpha \notin CN(\Delta')$, then Δ is α's minimal L-premise set. If there is no confusion, we will abbreviate L-premise set as premise set.

Definition 17. *$< \Gamma, \succ > | \sim_{S(L)} Gx(\alpha;\beta)$, iff.,*

(1) There is $\Delta \subseteq \Gamma, \Delta$ is the minimal premise set of $Gx(\alpha;\beta)$, and

(2) For any $\Lambda \subseteq \Gamma$, if Λ is the minimal premise set of $\neg Gx(\alpha;\beta)$, then $\Delta > \Lambda$, and

(3) For any $\Lambda \subseteq \Gamma$, if Λ is the minimal premise set of $Gx(\alpha; \neg\beta)$, then $\Delta > \Lambda$.

Definition 18. *Γ is any formula set. $CN(\Gamma)$ is the (generic) consequence set under priority order $\succ$, if $CN(\Gamma) = \{Gx(\alpha;\beta) :< \Gamma, \succ> \mid \sim_{S(L)} Gx(\alpha;\beta)\}$.*

Proposition 4. *Γ is any formula set. $\succ$ is the priority order on Γ. $CN(\Gamma)$ is reasonable.*

Let Γ is any formula set, for any finite formula set $\Delta \subseteq \Gamma$. $CN(\Delta)$ is a partial conclusion set reasoning from the premise set Γ. $CN(\Gamma)$ is the final-conclusion set reasoning (generic sentences) from the premise set.

1.4.3.2. Special priority order for getting generic sentences by deduction

Definition 19. *Let Γ is any formula set, $\succ_G$ is the G-priority order on Γ, iff., $\succ_G$ is the general priority order on Γ, which satisfies:*

(1) For any formula $\alpha, \beta, \gamma, \delta$, if $\forall x(\alpha \to \beta) \in \Gamma$, and $Gx(\alpha;\gamma)$, $Gx(\beta;\delta) \in \Gamma$, then $Gx(\alpha;\gamma) \succ_G Gx(\beta;\delta)$ (sub-category generics priority);

(2) For any formula α, β, if $\forall x(\alpha \to \beta) \in \Gamma$, and $Gx(\alpha;\gamma) \in \Gamma$, then $Gx(\alpha;\gamma) \succ_G \forall x(\alpha \to \beta)$ (generics priority).

From $\succ_G$, we can obtain the strict priority order on $\mathscr{P}(\Gamma)$ (denoted by $>_G$). From definitions 18 and definition 19, there is:

$$< \Gamma, \succ_G> \mid \sim_G Gx(\alpha;\beta)$$

This is the reasoning from Γ to $Gx(\alpha;\beta)$, which obtains generic sentences mainly through deduction, based on the logic G and the G-priority order on the premise set. If let G is G_D, then we can begin the GAG-type reasoning.

2 Quick review of the history of inductive reasoning

2.1 Induction

What is inductive reasoning? There is a definition on Wikipedia[5]: 'The premises of an inductive logic argument indicate some degree of support (inductive probability) for the conclusion but do not entail it; that is, they suggest truth but do not ensure it'. Wikipedia also says: 'though many dictionaries define inductive reasoning as reasoning that derives general principles from specific observations, this usage is out-dated'.

Now is the time to add some new ideas to this definition.

What is inductive reasoning? From a premise, we reach a conclusion only with some degree of support, which means that conclusions can be changed. This is non-monotonic reasoning.

Why do researchers use the qualifier 'degree' to define inductive reasoning? Let us review the history. There are two periods in the history of inductive reasoning: the classical period and the modern period. In the classical period, there are two representative researchers: Francis Bacon (1561–1626)and John Stuart Mill (1806–1873) tried to find methods to reach *certain* conclusions by induction. But in the modern period, started by John Maynard Keynes (1883–1946) who identified the first inductive logic, researchers no longer think that conclusions obtained by induction are certain. Based on the development of classic probability theory, researchers employ probability to express that uncertainty. Today, this idea and methodology are still in use. Due to the popular use of the qualifier 'degree', the concept of probability is employed to define uncertainty. However, probability is only one way to express uncertainty.

2.2 Core issues of inductive reasoning

Now, let us discuss the core issues of inductive reasoning. First, we express them in common usage.

1. How is the initial probability obtained?[6]

2. The logic system usually (after the initial probabilities) contains the axioms or rules for how to calculate probability.

3. Reasoning categories: inductive reasoning is used to reach conclusions about individuals; inductive reasoning is used to reach conclusions about classes.

[5]See in https://en.wikipedia.org/wiki/Inductive_reasoning.

[6]Some researchers, like Keynes, didn't provide their method for obtaining the initial probability.

Table 1 A brief review of the study on inductive reasoning

Periods	Representative researchers	Problem concerned	Technique used to deal with induction
Classical:	Bacon, F. (1561–1626), Mill, J. S. (1806–1873)	How to get *certain* conclusions by induction.	- Bacon's three tables - Mill's four canons
Modern:	Keynes, J. M. (1883–1946)	How to get *uncertain* conclusions by induction.	- Proposition logic + probability - First calculation of probability. - No method for obtaining initial probability.
	Reichenbach, H. (1891–1953)		- Predicate logic + probability - Frequency probability, obtain the initial probability by weight, weight posit
	Carnap, R. (1891–1970)		Modal logic + probability
	Cohen, L. J.		
	Burks, A. W.		
	...		

3 Comparing reasoning with generics and induction

We can now compare the main problems concerning inductive reasoning and reasoning with generics. If uncertain conclusions are represented as generic sentences, the problem above can be translated as follows:

1. How to obtain the generic sentence by induction.
2. Logic $M, G, G_D, \ldots$
3. Reasoning categories: the conclusion is a factual proposition; obtaining generic sentences is done mainly by deduction.

If we employ Reichenbach (1971)'s theory, things may become clearer. Reichenbach distinguished induction in primitive knowledge (primitive induction) and induction in advanced knowledge (advanced induction). Advanced induction is inductive reasoning with 'weights'.

In our theory, there is simple enumeration and inductive reasoning with weights: obtaining generic sentences mainly by induction and obtaining generic sentences mainly by deduction. The conclusion is a factual proposition.

4 Reasoning with generics versus probability

Sometimes we have to make a decision quickly, such as a weather forecast or a decision about an emergency, fire, or earthquake. This kind of reasoning is not within the scope of this study, and in my opinion, probability is the most helpful and practical way for that type type of inductive reasoning. However, what we are concerned with in this paper is reasoning about knowledge and beliefs, namely generic sentences. Table 2 compares these two ways of reasoning within the scope of this study.

Table 2 Comparing reasoning with generics and induction

	Generics	**Probability**
Expressiveness	intensive	extensive
Intuition	nature	
Scope	default reasoning, characterises human thinking process	statistics, optimal strategy
Application	?	$\surd$

5 Conclusion

Now, the conclusion is reached naturally. With generic reasoning, there is another way to interpret inductive reasoning: a method parallel to the probability method.

References

[Alc81] Alchourrón, C. E., & Makinson, D. (1981). Hierarchies of regulations and their logic. In R. Hilpinen (Ed.), *New studies in deontic logic* (pp. 125–148). Dordrecht: D. Reidel Publishing Company.

[Ash91] Asher, N., & Morreau, M. (1991). Commonsense entailment: A modal theory of nonmonotonic reasoning. In J. Mymopoulos & R. Reiter (Eds.), *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence* (pp. 387–392). Los Altos, CA: Morgan Kauffman.

[Ash95] Asher, N., & Morreau, M. (1995). What some generic sentences mean. In G. Carlson & F. Pelletier (Eds.), *The generic book* (pp. 300–338). Chicago: The University of Chicago Press.

[Coh99] Cohen, A. (1999). *Think generic!: The meaning and use of generic sentences*. Stanford, CA: Center for the Study of Language and Information (CSLI).

[Coh02] Cohen, A. (2002). Genericity, *Linguistische Berichte*, 59–89.

[Eck00] Eckardt, R. (2000). Normal objects, normal worlds and the meaning of generic sentences, *Journal of Semantics*, 16, 237–278.

[Kri95] Krifka, M., Pelletier, F. J., Carlson, G. N., Meulen, A., Chierchia, G., & Link, G. (1995). Genericity: An introduction. In G. Carlson & F. Pelletier (Eds.), *The generic book* (pp. 1–124). Chicago, IL: The University of Chicago Press.

[Mao03] Mao, Y. (2003). *A formalism for nonmonotonic reasoning encoded generics* (Unpublished doctoral dissertation). University of Texas, Austin, TX.

[Mao03] Mao, Y., & Zhou, B. (2003). An analysis of the meaning of generics, *Social Sciences in China*, XXIV:3, 126–133.

[Pel97] Pelletier, F. J., & Asher, N. (1997). Generics and defaults. In J. van Benthem & A. ter Meulen (Eds.), *Handbook of logic and language* (pp. 1125–1177). Cambridge, MA: MIT Press.

[Rei65] Reichenbach, H. (1965). *The theory of relativity and a priori knowledge*, Berkeley, CA: University of California Press.

[Rei71] Reichenbach, H. (1971). *The theory of probability: An inquiry into the logical and mathematical of the calculus of probability*, Berkeley, CA: University of California Press.

[Zha05] Zhang, L. (2005). *The analysis on the meaning of generics and one kind of reasoning encoded generics* (Unpublished doctoral dissertation). Peking University, Beijing.

[Zha09] Zhang, L. (2009). The reasoning encoded generics and orders on premise set, *Studies in Logic*, 2, 53–64.

[Zho08] Zhou, B. (2008). Sense semantics and term logics for generic reasoning. *Studies in Logic*, 1, 38–49.

[Zho03] Zhou, B., & Mao, Y. (2003). A basic logic for default reasoning. *Philosophical Researches*, supplement, 1–10.

[Zho04] Zhou, B., & Mao, Y. (2004). Default reasoning—The formal characterization for default reasoning. *Philosophical Trends*, supplement, 1–8.