Chapter 18
Knowledge Representation for Philosophers

Richmond H. Thomason

Abstract This article provides an overview of the subfield of Artificial Intelligence known as "Knowledge Representation and Reasoning." This field uses the techniques of philosophical logic, but aims at providing a theoretical basis for the management of declarative information in automated reasoning systems. Three topics are singled out here for attention: planning and reasoning about actions, description logics, and nonmonotonic logics.

18.1 Philosophical Logic and Logical AI

Formal philosophy seeks to use formalized languages and their metatheory to illuminate philosophical problems. In its earlier stages (roughly, until around 1960), most work in this area relied on classical logics and philosophical analysis, and so is difficult to distinguish from the broader area of analytic philosophy. But in its later stages, many practitioners of formal philosophy became convinced that classical logics were inadequate for some philosophical purposes, and the later work typically involves the formalization of a language, the development of its logical properties, and informal and philosophical discussion of its significance for philosophy.

A philosophical project of this sort uses what Alonzo Church ([13], pp. 47–58) called the *logistic method:* that is, it selects a target domain—an area of inquiry with characteristic forms of reasoning, and constructs a theory of the reasoning by providing a formalized logical system, including an axiomatization and model theory. Church had in mind mathematical domains and the sort of reasoning found in mathematical proofs, but philosophical logicians have used the method to study tense, modality, nondeclarative sentences, propositional attitudes such as knowledge and belief, contrary-to-fact conditionals, and many other linguistic constructions of philosophical interest.

R. H. Thomason (✉)
Philosophy Department, University of Michigan, Ann Arbor, MI, USA
e-mail: rthomaso@umich.edu

S. O. Hansson, V. F. Hendricks (eds.), *Introduction to Formal Philosophy*, Springer
Undergraduate Texts in Philosophy, https://doi.org/10.1007/978-3-319-77434-3_18

Artificial Intelligence is an eclectic field, and harbors many methodologies. But since the publication of [42], and largely because of John McCarthy's subsequent influence, the logistic method has been used to understand the domains that AI seeks to create. The targets include reasoning about time and action (and, in particular, planning), reasoning about other agents, about space and material objects, and many other topics.

Logical AI is continuous with earlier work in philosophical logic and makes explicit use of it; it is, in fact, best to think of philosophical logic and logical AI as a single field. Logical AI is now much larger and more active than its philosophical parent discipline, and by now many of the most important trends are being pursued by professional computer scientists. Nevertheless, most of this work is as relevant to philosophy as the earlier work that was published in the philosophical journals, and philosophers who value the usefulness of logic should be aware of the computational literature.

18.2 The Emergence of Knowledge Representation in AI

The process that led to the emergence of Knowledge Representation as a subfield of Artificial Intelligence should be of interest to philosophers. In both philosophy and AI formal techniques are available, but their value and appropriateness can be questioned. In AI, however, the foundational debate was limited to a few years, and resulted in a clearcut outcome. There are subfields of AI that can avoid reasoning about propositional attitudes and using formalisms that incorporate intensional constructions. But in those that do consider this sort of reasoning, the value of explicit representations, and of logical theory as a source of these representations, is no longer at issue.

The recognition of Artificial Intelligence as a field goes back to a conference held at Dartmouth College in 1956. (See [47].) A few years later, in [42], John McCarthy explicitly proposed an approach to AI that would attempt to represent an agent's declarative knowledge explicitly, employing logical rules as an inference mechanism. The need to formalize common-sense knowledge, and the appropriateness of logic for this purpose, is a continuing theme in McCarthy's later work; see the papers collected in [45].

However, McCarthy's early proposals were not very influential and, through the 1970s, much of the work in AI—and certainly, work that involved implemented systems—either ignored declarative representations entirely or, in some cases, developed ad hoc representation systems that had little or nothing to do with the logical tradition. (Marvin Minsky's "frame-based" representations are an example; see [49].)

During the phase of AI (roughly, dating through the 1970s and well into the 1980s), when researchers were concentrating on small to medium-sized reasoning problems, the role of logic was debated. See, for instance, [27, 48]; there is a retrospective discussion of the issues in [37, Section 1.5]. A thorough history of AI, and of the ideas in play during this period, remains to be written. But it is clear that during the latter part of the 1980s and the early 1990s, this conflict was decided in favor of the logicians—not so much by explicit debate, but by widespread recognition among AI practitioners of the importance and usefulness of logical representations.

I believe that the following factors played an important part in these developments:

1. **Software engineering considerations.** Beyond a certain size, it is difficult or impossible to maintain software systems without a modular design, and without a clear, explicit understanding of the meaning of the representations. As reasoning systems became larger and more ambitious, these considerations provided a powerful motivation for using logical representations when possible. As programs become larger and more complex, you need not only a comprehensive, detailed account of what the program is supposed to do; even better, you want a proof that the algorithm is correct.

   You also need modularity. The software engineering reasons for modular representation of declarative knowledge are well documented in [60, Chapter 3]. Stefik is eclectic about representation systems, but that brings me to my next point.
2. **Universality of logic for declarative representations.** Gradually, it became realized that various alternatives to logical representations that had been proposed could be formalized as logics, and that treating them as such would deliver improved insights.[1]
3. **Decoupling theories from implementations.** Theorem proving is seldom the best way to approach the reasoning problems that arise in AI. But, as the AI community learned, logical modeling doesn't commit an AI researcher to a theorem-proving implementation. Theorem proving is not the only algorithm associated with logic—for instance, model construction is useful for many purposes—and the relationship between a logical theory and an implementation informed by it can be tenuous. At one extreme, logical modeling helps to understand the reasoning problem, and although the implementation is inspired by the logic, it is hard to say what the relationship is. At another extreme, it can be hard to distinguish the theory from the implementation.
4. **Computer science graduate education.** Computer science began to produce graduate students in large numbers in the 1980s. As these students entered the AI research pool, the comfort level of AI researchers with logic grew. Computer science departments provide training in theory, and

[1] In [26], Patrick Hayes argued that frame-based representations, which had widely been taken to be an alternative to logical representations, could be reproduced in a first-order logic with a mechanism for formalizing defaults. (Hayes used an epistemic operator for this purpose.) The later history vindicated this idea, as ideas about frames and semantic nets were transformed into *description logics*—representation services that can be embedded in first-order logic, or in well understood extensions of first-order logic. See Sect. 18.4.2, below, for more about description logics.

theoretical computer science is almost entirely a branch of logic. Many of the younger AI researchers at this time were accomplished logicians.

5. **Small-scale successes.** Some early special-purpose uses of logic were successful and influential. Examples are James Allen's interval logic for reasoning about time [2], and McCarthy's Situation Calculus for reasoning about actions [43].

The first major collection devoted to knowledge representation, [9], appeared in 1985. At this point the field had begun to move rapidly, and many of the papers in the collection were already outdated, although the volume covers ideas that were to become important themes in the future. At this point, the area gained popularity: a significant number of papers devoted to knowledge representation began to appear at the major AI meetings.

A series of international conferences devoted entirely to knowledge representation began in Toronto in 1989; the twelfth in this series took place in 2010. By 1989, the field was well-established, and from now on I'll refer to it as "KRR". As we'll see, the second 'R' is important.

18.3 The Scope and Subject Matter of KRR

18.3.1 The Importance of Reasoning

The title of the first KR proceedings, [3], is "Principles of Knowledge Representation and Reasoning." The clause 'and reasoning' was added intentionally, and marks a significant difference between KRR and the closely related field of philosophical logic.

A typical project in philosophical logic will formalize some topic, hopefully providing a model-theoretic semantics as well as good motivation for the formalization. Usually, there are forms of reasoning associated with the domain that is formalized, and a philosophical logician will recognize this by taking into account examples of reasoning that intuitively are good or bad and using this to justify the validities delivered by the theory. Presumably, the intuitions about validity are closely associated with our expertise in the associated reasoning. But the connection of the project to reasoning doesn't go further than this. A philosophical logician will hope that the structures that make formulas true and false will deliver new insights into topics of traditional philosophical interest, and the philosophical impact of the project will mostly depend on the quality of thes insights. Except for a sample of valid and invalid inferences, reasoning is absent from these picture.

But many robust and complex forms of reasoning are associated with the domains that philosophical logicians have explored. For instance, consider the problem of reading a narrative and—if the narrative is temporally coherent—figuring out how to order the events that are mentioned in the narration into a temporal sequence, and maintaining this timeline as more of the narrative is read. Temporal logicians working in the philosophical logic tradition hardly ever consider issues of this sort.

But such questions are crucially important in knowledge representation, because their answers may provide connections to useful, implementable reasoning services.

Also—and this is a new consideration—the design of the formal language may be influenced by the intended reasoning application. In [35], Hector Levesque and Ronald Brachman argue that there is a potential tradeoff between expressive adequacy of the language and the computational complexity of the reasoning. A formalized language that is ambitious expressively may be less useful when the reasoning application is taken into account, because the reasoning associated with it—for instance, theorem proving or satisfaction checking—is more complex.

In fact, some of the most successful projects in KRR involve the discovery of useful compromises between the competing factors in the Brachman-Levesque tradeoff. Edmund Clarke's use in [14] of a restricted temporal language for software validation is an example, as well as the temporal language of [2]. But good solutions to the tradeoff are difficult to find. (The application to description logics that Levesque and Brachman had in mind did not quite work out as they had hoped.)

18.3.2 Topics in KRR

The coverage of the 12 KRR proceedings that appeared by 2010[2] provide a useful guide to the major topics, as well as an indication of their importance for the field. Among these topics are: Planning, Description logics, Abduction, Multiagent Systems, Nonmonotonic Logic, Planning Agents, Spatial Logics, Belief Revision, Ontologies, and Preferences. Not so well represented at the KRR meetings, but important for philosophers, is the work on formalizing common sense reasoning.

The next sections will go into more detail about a few of these topics; references to the others can be found in the works cited in Sect. 18.5.

18.4 Details About Selected KRR Topics

18.4.1 Planning and Reasoning About Actions

By any measure, the most active area of research in KRR consists of logics for planning. Planning itself, or means-end reasoning, was recognized quite early as an important area of AI (see [59]), and in the earliest phases the paradigmatic examples of planning were taken from gamelike domains. Simon's paper contains the fundamental idea that *actions* are available to the planning agent, which when executed will change the state of the world (for instance, the state of a partially

[2] These are [1, 3, 4, 11, 15, 16, 19–21, 23, 40, 51].

filled-in crossword puzzle), and the idea that planning is a matter of searching a problem space for a series of operators that will achieve a given *goal*.

It is certainly possible to implement a planning system without a logical formalization of the reasoning. (And many AI researchers did this, thinking only of the problem of how, using heuristic search, to efficiently find a plan in the very large search spaces that arise even in simple problems.) But some AI researchers, following John McCarthy, took a logical approach. McCarthy's Situation Calculus, first presented in [43] and mentioned above in Sect. 18.2, was offered as a formalization of means-end reasoning. The ideas in this paper (originally published in 1963) are elaborated in [46], which is usually cited as the source of the Situation Calculus.

There is a continuous history of research on the logic of action, within the Situation Calculus, from the early 1970s to the present, through which insights have deepened, and the theory has been generalized to more challenging planning domains.

Here, I will concentrate only on the basic ideas of the Situation Calculus formalism and on the immediate logical problems that it generates. For other expositions of this topic, see [39, 58, 62].

The Situation Calculus is a many-sorted first-order theory, with designated sorts for *situations* and for *actions*. ('Situation' is McCarthy's term for 'state'; actions are taken to be individuals.) Many predicates of the Situation Calculus, then, will have a single argument place of situational type: these are called *fluent predicates*, and the values they take in models are called *fluents*. Actions are treated as primitives: they are individuals, and there is a designated sort of actions.

There is a special 3-place predicate ***Result*** expressing a relation between situations, actions, and situations; the idea is that $\boldsymbol{Result}(s_1, a, s_2)$ is true iff performing the action denoted by a in the situation denoted by s_1 leads to the situation denoted by s_2. We will assume that the outcome of performing an action is unique:

$$\forall s \forall a \exists s_1 \forall s_2 [\boldsymbol{Result}(s, a, s_2) \leftrightarrow s_2 = s_1].$$

A *causal theory* in the Situation Calculus provides *causal axioms* for each action. A causal axiom for an action a is supposed to say what will happen if the action is performed in appropriate circumstances. At the very least, then, a causal axiom for a will entail a conditional relating a *precondition* for the action to its *effects*. The purpose of a causal theory is to characterize precisely what the result of performing each action will be. Without this, it would be impossible to tell in general what state would result from performing a series of actions, so it would be impossible to produce a plan supported by a proof that the goal will be reached.

In the crossword puzzle domain, for instance, for each letter and cell of the puzzle there is an action of putting that letter in the cell. We define an empty cell to be one that contains nothing:

$$\forall x \forall s [\boldsymbol{Empty}(x, s) \leftrightarrow \forall z \neg \boldsymbol{In}(z, x, s)].$$

Consider the action of putting ‘t’ in a cell. This action’s precondition is that the cell must be empty; its effect is that ‘t’ is the cell. The following simple causal axiom captures these things nicely:

$$\text{(SCA)}\ \forall s \forall x \forall s' \forall y [\boldsymbol{Result}(s, \boldsymbol{PutIn\text{-}t}, s') \rightarrow [\boldsymbol{Empty}(x, s) \rightarrow \boldsymbol{In}(\boldsymbol{t}, x, s')]].$$

(Here, $\boldsymbol{t}$ is a constant denoting the letter ‘t’.)

This causal axiom allows too many models. The problem is that it doesn’t say anything about what happens in cells other than the single cell that is affected. Of course, nothing happens in these other cells. We want *causal inertia* to prevail—the other cells stay put, but (SCA) doesn’t entail this.

The fact that causal inertia will in general call for many more axioms than are required for causal change is, in its purest form, the *Frame Problem*. In a more general form, the Frame Problem is the question of how to axiomatize causal inertia.[3]

A *monotonic* approach to the Frame Problem states the inertial rules explicitly as axioms. An economical way to do this is to write axioms giving necessary and sufficient conditions under which a fluent holds in an arbitrary resultant situation. In the crossword domain, the axiom for the letter ‘t’ would look like this:

$$\begin{aligned}\text{(MCA)}\ \forall s \forall a \forall s' \forall x [\, [\boldsymbol{Action}(a) \wedge \boldsymbol{Result}(s, a, s') \wedge \boldsymbol{In}(\boldsymbol{t}, x, s')] \leftrightarrow \\ [\, [a = \boldsymbol{PutIn\text{-}t} \wedge \boldsymbol{Empty}(x, s)] \vee [a \neq \boldsymbol{PutIn\text{-}t} \wedge \boldsymbol{In}(\boldsymbol{t}, x, s)]\,]\,].\end{aligned}$$

This axiom guarantees that ‘t’ appears in a cell of a noninitial situation if and only if it was already there, or was just put there. It requires quantification over actions, but this is unproblematic, since actions can (and probably should) be treated as individuals.

It’s a bit disappointing that monotonic solutions to the Frame Problem are perfectly workable (in [55], for instance, Ray Reiter hows how monotonic solutions can be deployed in challenging and complex planning environments), because the nonmonotonic solutions are so much more interesting from a logical standpoint. These solutions require a logic that somehow supports exceptions to axioms. (See Sect. 18.4.3, below.) Given such a logic, causal inertia can be expressed as a simple, global default: “nothing changes.” Causal axioms then provide constraints that override the inertial default.

The discovery of anomalies in the nonmonotonic solutions [25] led to increasingly complex logical solutions which, since they appeal to causality, are of considerable philosophical interest. See, for instance, [38, 63].

This is by no means the end of the logical challenges. The *Ramification Problem* has to do with indirect effects of actions; the *Qualification Problem* has to do with the difficulty of stating universally correct preconditions for actions. The

[3] Philosophers should take note. In the philosophical literature, the Frame Problem has been widely misunderstood and wildly overgeneralized. See [58, Section 1.12].

literature on both these problems is extensive, and contains much material that is philosophically interesting; some entry points to the literature are [38, 41, 61].

Many other problems arise in the process of extending simple planning theories to more noisy and challenging domains. What about multiple agents? What about agents who are uncertain about the current state of the world? How can the discrete, action-based accounts of change used in planning be combined with continuous theories of natural change based on differential equations? Many authors have discussed these issues, but [55] is perhaps the best beginning point.

Before turning to other matters, let's consider how the logical approach to planning allows declarative information (knowledge) to be separated from the heuristics and procedures that may be involved in reasoning with the knowledge. The causal axioms, other domain axioms, and the causal theory constitute a declarative theory that can be formalized in a logic with well understood properties. It can be validated by checking it against intuitions and evidence about the domain. It is relatively easy to update. And it is independent of any particular implementation. It can be combined with any algorithm for finding a plan, and with any heuristics for narrowing the search, that an implementer chooses to use. Each of these things is an advantage, from a software engineering standpoint. Taken together, the case for this modular approach, separating out the declarative knowledge and using a logic to formalize it, is quite compelling.

18.4.2 Description Logics

Many AI applications, as well as planning, will need a separate representation of the knowledge used by the system. This creates a need for a plug-in KR service that is relatively easy to learn and to use, and that can reliably and efficiently deliver the conclusions that are needed by the system. Description logics fill this niche better than any other KR service.

There are many description logics, so I will confine myself here to the basic recipe for a description logic: separate general from factual information. Insist that general information takes the form of concept definitions, and include in the KR language useful constructs for forming such definitions. Restrict the form of the factual information; for instance, do not allow disjunctive formulas.

Definitions in a description logic might look like this:

Mother : HAS-AT-LEAST-1(***child***) AND ***Female***.
Employed : HAS-AT-LEAST-1(***employer***).
Working-Mother : ***Mother*** AND ***Employed***.
Orphan : HAS-0(***parent***).
parent : INVERSE(***child***).

Given these definitions, a description logic would be able to infer that Agnes is a working mother if it is told that Agnes is a mother and is self-employed. You would hope that it would be able to infer an inconsistency if it is told that Bert is Agnes' child and Bert is an orphan.

The reasoning algorithms for many description logics are well understood and deliver reliable results, often with excellent efficiency. Good documentation is available for many of these systems. Much work has been devoted to extending the expressiveness of description logics, and many of these extensions—for instance, attempts to include temporal reasoning—are philosophically interesting.

See [6] for a recent survey of this topic, with many references; [5] has many details, including descriptions of some of the leading systems.

18.4.3 Nonmonotonic Logics and Nonmonotonic Reasoning

Nonmonotonic logic might well have been developed earlier, by philosophical logicians, but in fact this topic emerged from logical AI. Monotonicity is a property of the consequence relation $\vdash$: if $\Gamma \vdash \phi$ then $\Gamma \cup \{\psi\} \vdash \phi$. This says that adding a new axiom to an axiomatic system produces more theorems. A consequence relation is nonmonotonic if it fails to have the monotonicity property.

Common-sense reasoning is full of examples of nonmonotonicity. For instance, let Γ be the set of observations that assumptions that are in play for for a practical agent—a person or a robot, and let Δ be the set of conclusions that the agent draws from these observations. Suppose the agent observes, in situation s_1, that a certain cup is on a certain table. Then the formula

$$\text{(A1) } \boldsymbol{On}(\boldsymbol{cup87}, \boldsymbol{table15}, s_1)$$

will be in Γ. The agent has no reason to think the cup will be disturbed in the span of time between s_1 and, say, s_2. Then our agent will suppose (A2), which will therefore belong to Δ, because it is concluded from observations and the agent's causal theory.

$$\text{(A2) } \boldsymbol{On}(\boldsymbol{cup87}, \boldsymbol{table15}, s_2)$$

Suppose the agent daydreams, receiving no new information between s_1 and s_2. Observing the table, the agent learns that, contrary to expectations, the cup is gone; the new observation (A3) is the negation of (A2).

$$\text{(A3) } \neg\boldsymbol{On}(\boldsymbol{cup87}, \boldsymbol{table15}, s_2)$$

In a monotonic logic, we get an inconsistent theory if we add (A3) to Γ. In a nonmonotonic logic, the conclusion is retracted when the addition is made and $\Gamma \cup \{(A3)\}$ is consistent. The generalization that produced the incorrect conclusion—

in this case, a nonmonotonic axiom of causal inertia—is retained. The conclusion that is withdrawn marks an exception to the axiom.

John McCarthy's Circumscription Theory, proposed in [44], involves a relatively simple modification of first-order logic. Since it can be fairly easily explained, we will use it to illustrate the workings of a nonmonotonic logic. The language of Circumscription Theory is first-order, but special *abnormality* predicates are introduced to mark exceptions.

An exception-tolerant Causal Inertia axiom for the crossword puzzle domain would be stated as follows.

$$\text{(NMCI)}\ \forall x \forall y \forall s \forall a[[\boldsymbol{Cell}(x) \wedge \boldsymbol{In}(y, x, s) \wedge \boldsymbol{Result}(s, a, s') \wedge \neg \boldsymbol{Ab}(x, y, s)] \rightarrow \\ \boldsymbol{In}(y, x, s')$$

The axiom guarantees that what is in a cell stays put through a change unless there is an abnormality involving the cell, its occupant, and the change. (Another inertia axiom would be needed to ensure that empty cells stay empty unless there is an exception.)

We obtain a nonmonotonic logic by taking account in the definition of logical consequence only models of a theory Γ in which the extensions of the various abnormality predicates are minimized. With just one abnormality predicate, say the 3-place predicate in (NMCI), the definition is simple. A model M is *better* than a model M', $M \prec M'$, iff the extension Ab of $\boldsymbol{Ab}$ in M is a proper subset of the extension Ab$'$ of $\boldsymbol{Ab}$ in M'. A model M of Γ is *minimal* iff no model of Γ is better than M. Finally $\Gamma \vdash \phi$ iff every minimal model of Γ satisfies ϕ.

One advantage of Circumscription, from the standpoint of formalizing domains, is that it is possible to write axioms about abnormalities—about what to expect when things go wrong. These axioms themselves may involve abnormalities. See [36] for details. This article is also an excellent introduction to Circumscription Theory; and treatments can be found in any of the books on nonmonotonic logic cited in the bibliography to this paper, as well as in [12].

A very large body of work on nonmonotonic logic and its applications has accumulated over the last 30 years, most of it appearing in the AI journals, but some in philosophical venues.

The two other leading approaches to nonmonotonic logic are *Default Logic* and modal theories such as *Autoepistemic Logic*. Default Logic, originating in [54], takes a more proof-theoretic approach to the topic. A default theory consists of two components: the monotonic component is a set of ordinary first-order axioms, and the nonmonotonic component is a set of *default rules*.

In the special case I'll consider here,[4] a (normal) default rule

$$\phi \ / \ \psi$$

[4] I will ignore general default rules in this exposition, and only consider normal defaults.

looks like an ordinary rule of inference, but has a different interpretation: the conclusion ψ can be inferred from the premise ϕ as long as it is consistent to do so.

A default rule like

$$\text{(DR1)}\ \boldsymbol{\mathit{TurnSwitch}}\ /\ \boldsymbol{\mathit{LightOn}}$$

could be read "Infer that the light will go on if you turn the switch."

If the monotonic axioms are $\{\boldsymbol{\mathit{TurnSwitch}}\}$ and the only default rule is (DR1), then $\boldsymbol{\mathit{LightOn}}$ can be concluded. But if the monotonic axioms are $\{\boldsymbol{\mathit{TurnSwitch}}, \neg\boldsymbol{\mathit{LightOn}}\}$, then $\boldsymbol{\mathit{LightOn}}$ cannot be concluded.

Just as the rules of first-order logic allow a theory to be derived from first-order axioms, Reiter assumes that *extensions* can be derived from a default theory: an extension is a set of formulas that a perfect reasoner might infer from the monotonic axioms and default rules of the theory.

But defaults can *conflict*, and this makes things complicated. The standard example is the *Nixon Diamond*: the monotonic theory is $\{\boldsymbol{\mathit{Quaker}}(\boldsymbol{\mathit{Nixon}}),\boldsymbol{\mathit{Republican}}(\boldsymbol{\mathit{Nixon}})\}$ and there are two default rules:

$$\boldsymbol{\mathit{Quaker}}(\boldsymbol{\mathit{Nixon}})\ /\ \boldsymbol{\mathit{Pacifist}}(\boldsymbol{\mathit{Nixon}})\ \text{and}\ \boldsymbol{\mathit{Republican}}(\boldsymbol{\mathit{Nixon}})\ /\ \neg\boldsymbol{\mathit{Pacifist}}(\boldsymbol{\mathit{Nixon}}).$$

Here, it is not clear what to say: both default rules can be consistently applied separately to the monotonic axioms, but not both. Reiter associates *two* extensions with the Nixon Diamond: one concludes that Nixon is a pacifist, the other that Nixon is not a pacifist. Given the information in the default theory, and forgetting whatever else we know about Nixon, there is no way to choose between these two extensions.

Theorists differ about how to think about this, but the most interesting interpretation, from a logical standpoint, is that the relation between premises and their logical consequences is not unique in nonmonotonic logic: perfect reasoners can draw different sets of conclusions from the same default theory. This idea is particularly attractive in metaethical applications of nonmonotonic logic; see John F. Horty's work, cited in the bibliography.

Reiter's main technical achievement in [54] consists of two definitions of the extension relation, and a proof that the definitions are equivalent. For more about default logic, see [7, 18, 32, 57], and any of the general treatments of nonmonotonic logic listed in the bibliography.

The thought behind autoepistemic logic is that a default rule applies unless the reasoning agent knows something to the contrary; this suggests that defaults can be formalized using an epistemic modal operator. For more about this approach, see [33] and (again) any of the general references to nonmonotonic logic in the bibliography.

The field of *Argumentation Theory* is only tenuously connected to KR, but it uses ideas from nonmonotonic logic and is potentially important for philosophy. The idea is to treat arguments abstractly, constructing a theory of notions like the relations of *attack* and *defeat* between arguments, and attempts to develop a notion of *extension* for arguments analogous to Reiter's extensions for default logic. The literature on Argumentation Theory is by now rather extensive. It has been applied to many reasoning domains, including the law, but has not yet gotten the attention it deserves from philosophers.[5]

For general discussions of Argumentation Theory. see [8, 53]. [22] is an early, influential paper in the field. For applications to the law, see, [34, 56].

Although I will only mention the programming language PROLOG and its extensions briefly here, there is a strong, continuous tradition of work in this area in KRR. With its declarative programming style, PROLOG programming offers a distinctive and important compromise between declarative transparency and implementability. PROLOG's negation-as-failure provides a connection to nonmonotonic reasoning.

And PROLOG can be extended in interesting ways. Some of these extensions become large-scale projects that attract research groups, and offer KRR services for important areas of reasoning. I have already mentioned Ray Reiter's extension in [55] of PROLOG into a language for cognitive robotics. Stable models and answer sets are another area of this kind; see [24].

Horty's work combining nonmonotonic logic and deontic logic provides a good example of how ideas originating in KRR can be fruitfully applied in metaethics; see [28–31]. Certainly, these ideas can be applied in many other areas of philosophy as well.

18.5 How Can a Philosopher Access the Field?

Much of the literature in KRR is technical, but this should not be a problem for formal philosophers—especially since so much of it overlaps with philosophical logic. For those who want a systematic introduction, [10] is an excellent resource. For those interested in specific topics, as well as references to the literature, [64] is a very useful resource. For commonsense reasoning, see [17, 50]. If anyone wants to get a comprehensive, detailed sense of the research in this field, there is nothing like the KRR proceedings listed above in Footnote 2. There is a great deal of material there, but the quality is very high.

Bibliography

1. Aiello, L. C., Doyle, J., & Shapiro, S. (Eds.). (1996). *KR'96: Principles of Knowledge Representation and Reasoning*. San Francisco: Morgan Kaufmann.
2. Allen, J. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM, 26*(11), 832–843.

[5] John Pollock's work, however, is an exception. Pollock developed a theory of nonmonotonic reasoning that is closely related to Argumentation Theory. See, for instance, [52].

3. Allen, J. F., Fikes, R., & Sandewall, E. (Eds.). (1989). *KR'89: Principles of Knowledge Representation and Reasoning*. San Mateo: Morgan Kaufmann.
4. Allen, J. F., Kautz, H. A., Pelavin, R., & Tennenberg, J. (Eds.). (1991). *KR'91: Principles of Knowledge Representation and Reasoning*. San Mateo: Morgan Kaufmann.
5. Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D., & Patel-Schneider, P. (Eds.). (2003). *The description logic handbook: Theory, implementation and applications*. Cambridge: Cambridge University Press.
6. Baader, F., Horrocks, I., & Sattler, U. (2008). Description logics. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of knowledge representation* (pp. 135–179). Amsterdam: Elsevier.
7. Besnard, P. (1992). *Default logic*. Berlin: Springer.
8. Besnard, P., & Hunter, A. (2008). *Elements of argumentation*. Cambridge, MA: The MIT Press.
9. Brachman, R. J., & Levesque, H. J. (Eds.). (1985). *Readings in knowledge representation*. Los Altos: Morgan Kaufmann.
10. Brachman, R. J., & Levesque, H. (2004). *Knowledge representation and reasoning*. Amsterdam: Elsevier.
11. Brewka, G., & Lang, J. (Eds.). (2008). *KR2008: Proceedings of the Eleventh International Conference*. Menlo Park: AAAI Press.
12. Brewka, G., Niemelä, I., & Truscyński, M. (2008). Nonmonotonic reasoning. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of knowledge representation* (pp. 239–284). Amsterdam: Elsevier.
13. Church, A. (1959). *Introduction to mathematical logic* (Vol. 1). Princeton: Princeton University Press.
14. Clarke, E. M., Grumberg, O., & Peled, D. A. (1999). *Model checking*. Cambridge, MA: The MIT Press.
15. Cohn, A. G., Schubert, L., & Shapiro, S. C. (Eds.). (1998). *KR'98: Principles of Knowledge Representation and Reasoning*. San Francisco: Morgan Kaufmann.
16. Cohn, A. G., Giunchiglia, F., & Selman, B. (Eds.). (2000). *KR2000: Principles of Knowledge Representation and Reasoning*. San Francisco: Morgan Kaufmann.
17. Davis, E. (1991). *Representations of common sense knowledge*. San Francisco: Morgan Kaufmann.
18. Delgrande, J. P., & Schaub, T. (2000). The role of default logic in knowledge representation. In J. Minker (Ed.), *Logic-based artificial intelligence* (pp. 107–126). Dordrecht: Kluwer Academic Publishers.
19. Doherty, P., Mylopoulos, J., & Welty, C. A. (Eds.). (2006). *KR2006: Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning*. Menlo Park: AAAI Press.
20. Doyle, J., Sandewall, E., & Torasso, P. (Eds.). (1994). *KR'94: Principles of Knowledge Representation and Reasoning*. San Francisco: Morgan Kaufmann.
21. Dubois, D., Welty, C., & Williams, M.-A. (Eds.). (2004). *KR2004: Principles of Knowledge Representation and Reasoning*. AAAI Press.
22. Dung, P. M. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. *Artificial Intelligence, 77*(2), 321–257.
23. Fensel, D., Giunchiglia, F., McGuinness, D., & Williams, M.-A. (Eds.). (2002). *KR2002: Principles of Knowledge Representation and Reasoning*. San Francisco: Morgan Kaufmann.
24. Gelfond, M. (2008). Answer sets. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of knowledge representation* (pp. 285–316). Amsterdam: Elsevier.
25. Hanks, S., & McDermott, D. (1986). Default reasoning, nonmonotonic logics and the frame problem. In T. Kehler & S. Rosenschein (Eds.), *Proceedings of the Fifth National Conference on Artificial Intelligence* (pp. 328–333), Los Altos. Morgan Kaufmann: American Association for Artificial Intelligence.
26. Hayes, P. (1981). The logic of frames. In B. Webber & N. J. Nilsson (Eds.), *Readings in artificial intelligence* (pp. 451–458). Los Altos: Morgan Kaufmann.

27. Hayes, P. (1987). A critique of pure reason. *Computational Intelligence, 3*, 179–185.
28. Horty, J. F. (1994). Moral dilemmas and nonmonotonic logic. *Journal of Philosophical Logic, 23*(1), 35–65.
29. Horty, J. F. (1995). Deontic logic and nonmonotonic reasoning. In D. Nute (Ed.), *Essays in defeasible deontic logic*. Dordrecht: Kluwer Academic Publishers.
30. Horty, J. (1997). Nonmonotonic foundations for deontic logic. In D. Nute (Ed.), *Defeasible deontic logic* (pp. 17–44). Dordrecht: Kluwer Academic Publishers.
31. Horty, J. F. (2001). *Agency and deontic logic*. Oxford: Oxford University Press.
32. Horty, J. F. (2007). Defaults with priorities. *Journal of Philosophical Logic, 36*(4), 367–413.
33. Konolige, K. (1994). Autoepistemic logic. In D. Gabbay, C. J. Hogger, & J. A. Robinson (Eds.), *Handbook of logic in artificial intelligence and logic programming, volume 3: Nonmonotonic reasoning and uncertain reasoning* (pp. 217–295). Oxford: Oxford University Press.
34. Kowalski, R. A., & Toni, F. (1996). Abstract argumentation. *Artificial Intelligence and Law, 4*, 275–296.
35. Levesque, H. J., & Brachman, R. J. (1995). A fundamental tradeoff in KR and reasoning. In R. J. Brachman & H. J. Levesque (Eds.), *Readings in knowledge representation*. Los Altos: Morgan Kaufmann.
36. Lifschitz, V. (1988). Circumscriptive theories: A logic-based framework for knowledge representation. *Journal of Philosophical Logic, 17*(3), 391–441.
37. Lifschitz, V., Morgenstern, L., & Plaisted, D. (2008). Knowledge representation and classical logic. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of knowledge representation* (pp. 3–88). Amsterdam: Elsevier.
38. Lin, F. (1995). Embracing causality in specifying the indirect effects of actions. In C. Mellish (Ed.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (pp. 1985–1991). San Francisco: Morgan Kaufmann.
39. Lin, F. (2008). Situation calculus. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of knowledge representation* (pp. 649–669). Amsterdam: Elsevier.
40. Lin, F., Sattler, U., & Truszczynski, M. (Eds.). (2010). *KR2010: Proceedings of the Twelfth International Conference*. Menlo Park: AAAI Press.
41. McCain, N., & Turner, H. (1995). A causal theory of ramifications and qualifications. In C. Mellish (Ed.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (pp. 1978–1984). San Francisco: Morgan Kaufmann.
42. McCarthy, J. (1959). Programs with common sense. In *Proceedings of the Teddington Conference on the Mechanization of Thought Processes* (pp. 75–91), London. Her Majesty's Stationary Office.
43. McCarthy, J. (1969). Situations, actions, and causal laws. In M. Minsky (Ed.), *Semantic information processing* (pp. 410–417). Cambridge: The MIT Press. Originally published in 1963 as a technical report.
44. McCarthy, J. (1980). Circumscription—a form of non-monotonic reasoning. *Artificial Intelligence, 13*(1–2), 27–39.
45. McCarthy, J. (1990). *Formalizing common sense: Papers by John McCarthy*. Norwood: Ablex Publishing Corporation. Edited by Vladimir Lifschitz.
46. McCarthy, J., & Hayes, P. J. (1969). Some philosophical problems from the standpoint of artificial intelligence. In B. Meltzer & D. Michie (Eds.), *Machine intelligence 4* (pp. 463–502). Edinburgh: Edinburgh University Press.
47. McCarthy, J., Minsky, M. L., Rochester, N., & Shannon, C. E. (2006). A proposal for the Dartmouth summer research project on artificial intelligence. *The AI Magazine, 27*(4), 12–14.
48. McDermott, D. (1987). A critique of pure reason. *Computational Intelligence, 3*, 151–160.
49. Minsky, M. (1981). A framework for representing knowledge. In J. Haugeland (Ed.), *Mind design* (pp. 95–128). Cambridge, MA: The MIT Press. Originally published in 1974 as an MIT technical report.
50. Mueller, E. T. (2006). *Commonsense reasoning*. Amsterdam: Elsevier.
51. Nebel, B., Rich, C., & Swartout, W. (Eds.). (1992). *KR'92: Principles of Knowledge Representation and Reasoning*. San Francisco: Morgan Kaufmann.

52. Pollock, J. L. (2001). Defeasible reasoning with variable degrees of justification. *Artificial Intelligence, 133*(1–2), 233–282.
53. Prakken, H., & Vreeswijk, G. (2001). Logics for defeasible argumentation. In D. M. Gabbay & F. Guenthner (Eds.), *Handbook of philosophical logic, volume IV* (2nd ed., pp. 219–318). Amsterdam: Kluwer Academic Publishers
54. Reiter, R. (1980). A logic for default reasoning. *Artificial Intelligence, 13*(1–2), 81–132.
55. Reiter, R. (2001). *Knowledge in action: Logical foundations for specifying and implementing dynamical systems*. Cambridge, MA: The MIT Press.
56. Rissland, E. L., Ashley, K. D., & Loui, R. P. (2003). AI and law: A fruitful synergy. *Artificial Intelligence, 150*(1–2), 1–15.
57. Schaub, T. (1998). The family of default logics. In D. M. Gabbay & P. Smets (Eds.), *Handbook of defeasible reasoning and uncertainty management systems, volume 2: Reasoning with actual and potential contradictions* (pp. 77–134). Dordrecht: Kluwer Academic Publishers.
58. Shanahan, M. (1997). *Solving the frame problem*. Cambridge, MA: The MIT Press.
59. Simon, H. A. (1966). *On reasoning about actions* (Technical Report Complex Information Processing Paper #87), Carnegie Institute of Technology, Pittsburgh.
60. Stefik, M. J. (1995). *An introduction to knowledge systems*. San Francisco: Morgan Kaufmann.
61. Thielscher, M. (2001). The qualification problem: A solution to the problem of anomalous models. *Artificial Intelligence, 131*(1–2), 1–37.
62. Thomason,R. H. (2003). Logic and artificial intelligence. Stanford Encyclopedia of Philosophy. http://plato.stanford.edu/archives/fall2003/entries/logic-ai/.
63. Turner, H. (2008). Nonmonotonic causal logic. In F. van Harmelen, V. Lifschitz, & B. Porter (Eds.), *Handbook of knowledge representation* (pp. 759–776). Amsterdam: Elsevier.
64. van Harmelen, F., Lifschitz, V., & Porter, B. (Eds.). (2008). *Handbook of knowledge representation*. Amsterdam: Elsevier.