

Reasoning in Non-Axiomatic Logic: A Case Study in Medical Diagnosis

Pei Wang and Seemal Awan

Temple University, Philadelphia, PA 19122, USA
{pei.wang, seemal.awan}@temple.edu

Abstract. Non-Axiomatic Logic (NAL) is designed for intelligent reasoning, and can be used in a system that has insufficient knowledge and resources with respect to the problems to be solved. This paper reports the result of a case study that applies NAL in medical diagnostics, and the logic is compared with binary logic and probability theory.

Keywords: reasoning, uncertainty, learning, medical diagnosis

1 Problem and Background

In a broad sense, “reasoning” is a cognitive process in which new pieces of knowledge (or beliefs) are derived from existing ones. In this process, the input, output, and intermediate results are specified as sentences of a language in which the words correspond to concepts. The process consists of a sequence of steps, each of which is an instance of a general pattern (rule or schema) that has justification. Therefore, a reasoning system can be considered as following a “logic,” which consists of the following components:

- Language**, which specifies the format or pattern of the sentences recognizable and acceptable by the system,
- Rules**, which describes the form of conclusions that can be derived from given premises in the system,
- Semantics**, which provides interpretation for the language and justification for the rules.

A system following or implementing such a logic also needs a memory to store the knowledge and to provide a working space, as well as a control strategy to select the rule(s) and premise(s) of each step. For a reasoning system to solve practical problems, domain knowledge should also be provided.

From the viewpoint of Artificial General Intelligence (AGI), reasoning systems are interesting not only because reasoning is arguably a necessary capability of any intelligent system, but also because such a system provides a clear separation between the domain-independent *design* of the system (that is, the logic and control mechanism) and its domain-specific *content* (the knowledge). The design is “general-purpose”, since the system can be given different knowledge to gain expertise in various domains, without changing the design.

This idea is not new to AI. The first wave of practical applications that made AI an industry was the knowledge-based expert systems [8]. However, though such expert systems have been successfully built for certain domains, the techniques have not grown to other domains as expected. Among the issues raised, *robustness* and *scalability* are prominent; that is, most of the expert systems fail to deal with unexpected situations with affordable time-space resources.

A major reason of this failure may be found in the theoretical foundation of these systems. At the current time, the two major theories on reasoning are *mathematical logic* and *probability theory*. Mathematical logic [3, 17] is often applied in AI the form of non-monotonic logic [4] or description logic [2]. Probability theory and statistics have also been used in AI as a model of reasoning, as in Bayesian Network [6]. Though both theories have achieved great successes in many fields, neither of them was designed to capture all major aspects of human reasoning. Mathematical logic was created to provide a theoretical foundation for mathematics, so it focuses on the type of inference used in proving mathematical assertions — symbolic binary deduction that derives theorems from a set of axioms or postulates. Probability theory models uncertainty in reasoning by treating the degree of belief as a probability distribution over a closed belief space, and reasoning on this space is carried out according to the axioms of probability theory. In both theories, the conclusions are restricted by the initial assumptions, and the derivation process may demand resources that are not affordable in practical applications.

What we want are reasoning systems that are not only justifiable according to certain rational principles, but also work in realistic situations, by using available knowledge and resources to derive the best conclusions the system can get. We hope that the capability and performance of the system will be comparable to that of a human being, though it is not necessary (or even desired) for the concrete behaviors of the system to be identical to that of the human beings.

2 NAL Overview

Non-Axiomatic Logic, or NAL, was designed to be a logic system that can be used when a system has *insufficient knowledge and resources*, that is, the perfect solution to a problem is beyond the knowledge scope and resource capacity of the system [9, 13]. In such a situation, a *rational* solution is the one that is best supported by the available evidence that the system can find with the available resources. Since this logic has been described in previous publications (see the first author's website¹), in this paper it is only briefly introduced.

Like other modern logical systems, NAL uses a formal language, *Narsese*, to represent its knowledge. In Narsese, each *term* is the name of a concept. In the simplest case, we can use English nouns or noun phrases for terms, such as “*patient*” or “*flu-patient*.” Unlike conventional reasoning systems, where the meaning of a concept is taken to be the objects in the world it refers to, the

¹ <http://www.cis.temple.edu/~pwang/>

meaning of a term in NAL is determined by what the system knows about it, that is, its conceptual relations with other terms that have been experienced by the system.

In Narsese, the most basic conceptual relation is *inheritance*, symbolized as “ \rightarrow .” For example, “Flu-patient is a type of patient.” is expressed as the statement “*flu-patient* \rightarrow *patient*,” where “*flu-patient*” is the *subject term*, and “*patient*” is the *predicate term*. In general, an inheritance statement states that the subject is a *special case* of the predicate and the predicate is a *general case* of the subject; or, in other words, the subject represents certain *instances* of the predicate, and the predicate represents certain *properties* of the subject.

To represent an individual instance (corresponding to a proper noun in English) and an elementary property (corresponding to an adjective in English), an *extensional set* and *intensional set* are used, respectively. For example, “John is a patient” is represented in Narsese as “ $\{John\} \rightarrow patient$,” and “Patients are sick” as “*patient* $\rightarrow [sick]$.” Here, terms like $\{John\}$ and $[sick]$ are not *simple* terms anymore, but rather *compound* terms, formed by certain operators from other terms. Other compound terms correspond to the *intersection*, *union*, *difference*, etc., of terms, such as $(doctor \cap patient)$ (“doctor and patient”), $(doctor \cup patient)$ (“doctor or patient”), and $(doctor - patient)$ (“doctor but not patient”).

If the relation between terms A and B cannot be directly represented as inheritance or its variants, but a term R whose meaning is empirically decided, then in Narsese it can be expressed as “ $(A \times B) \rightarrow R$,” with a compound term as subject. Intuitively, it states that “The relation from A to B is a type of R .” For example, “John is Mary’s son” can be expressed as “ $(\{John\} \times \{Mary\}) \rightarrow son-of$.” The same sentence can also be expressed as “ $\{John\} \rightarrow (\perp son-of \diamond \{Mary\})$ ” and “ $\{Mary\} \rightarrow (\perp son-of \{John\} \diamond)$ ”, where the symbol ‘ \diamond ’ indicates the location of the subject in the relation.

Statements are also defined as compound terms, and using them, Narsese can represent very complicated sentences. The complete grammar of Narsese can be found at [13] and the project website.

Assuming insufficient knowledge, in NAL “truth” is a matter of degree, indicating the evidential support a statement gets from available evidence. For statement “*flu-patient* \rightarrow *patient*,” a common instance or property of the two terms provides a piece of *positive* evidence, since as far as it is considered, the statement is true. On the other hand, if an instance of “*flu-patient*” is not an instance of “*patient*,” or a property of “*patient*” is not a property of “*flu-patient*,” it is *negative* evidence for the statement.

For a statement, if the amounts of *positive* and *negative* evidence are measured by real numbers w^+ and w^- , respectively, then the ratio $w^+/(w^+ + w^-)$ naturally represents one aspect of the uncertainty of the statement, that is, the proposition of available evidence that supports the statement. This ratio is called “frequency” in NAL. Since new evidence comes into the system from time to time, a frequency value may change over time. To measure the stability of a frequency value, the amount of *available evidence* w ($w = w^+ + w^-$) is compared to

a constant amount of *future evidence* k (with 1 as the default value), and the ratio $w/(w+k)$ is called the “confidence” of the statement (so this word is used differently from the “confidence interval” in statistics). The $\langle \textit{frequency}, \textit{confidence} \rangle$ pair forms the *truth-value* of a statement in NAL. Defined as above, a truth-value is not determined according to whether the statement corresponds to a fact in a model, but what the system knows about the statement. Together with the previous definition of meaning, this definition of truth forms the core of the *experience-grounded semantics* of NAL, which is fundamentally different from the *model-theoretic semantics* used in traditional logical systems.[13]

When this logic is applied to a practical situation, the truth-value of the knowledge initially given to the system is determined by the user according to the above semantics. If the knowledge comes from statistical data, then the amount of evidence can be directly measured as the sample size, which in turn decides the truth-value. If the knowledge comes in qualitative form, conventions are used to assign quantitative truth-values. For example, in the current implementation a normal affirmative sentence gets the default truth-value $\langle 1.0, 0.9 \rangle$, which correspond to $w^+ = 9$ and $w^- = 0$.

According to experience-grounded semantics, each inference rule should have an associated truth-value function to determine the truth-value of the conclusion according to the type of inference and the truth-values of the premises. This is the case because in each inference step, the evidence of the conclusion comes from the premises only, and the other knowledge in the system is not directly involved. The design of the NAL truth-value functions is discussed in [9, 13] and other previous publications on the project, so in the following we only list a few typical rules with their truth-value functions, without explaining why they are designed in the current form.

The *deduction* rule specifies how the *transitivity of inheritance* is extended into multi-value statements. It takes “ $M \rightarrow P \langle f_1, c_1 \rangle$ ” and “ $S \rightarrow M \langle f_2, c_2 \rangle$ ” as the premises, and derives “ $S \rightarrow P \langle f_1 f_2, f_1 f_2 c_1 c_2 \rangle$ ” as the conclusion. So, given “ $\textit{patient} \rightarrow [\textit{sick}] \langle 1, 0.90 \rangle$ ” and “ $\{\textit{John}\} \rightarrow \textit{patient} \langle 1, 0.90 \rangle$,” the rule derives “ $\{\textit{John}\} \rightarrow [\textit{sick}] \langle 1, 0.81 \rangle$ ” (“John is sick”).

The *induction* rule evaluates an inheritance statement by checking a *common instance* of the two terms. It takes “ $M \rightarrow P \langle f_1, c_1 \rangle$ ” and “ $M \rightarrow S \langle f_2, c_2 \rangle$ ” as the premises, and derives “ $S \rightarrow P \langle f_1, f_2 c_1 c_2 / (f_2 c_1 c_2 + k) \rangle$ ” as the conclusion. Given “ $\{\textit{John}\} \rightarrow [\textit{sick}] \langle 1, 0.90 \rangle$ ” and “ $\{\textit{John}\} \rightarrow \textit{patient} \langle 1, 0.90 \rangle$,” the rule derives “ $\textit{patient} \rightarrow [\textit{sick}] \langle 1, 0.45 \rangle$ ” (“Patients may be sick”).

The *abduction* rule evaluates an inheritance statement by checking a *common property* of the terms. It takes “ $P \rightarrow M \langle f_1, c_1 \rangle$ ” and “ $S \rightarrow M \langle f_2, c_2 \rangle$ ” as the premises, and derives “ $S \rightarrow P \langle f_2, f_1 c_1 c_2 / (f_1 c_1 c_2 + k) \rangle$ ” as the conclusion. Given “ $\textit{patient} \rightarrow [\textit{sick}] \langle 1, 0.90 \rangle$ ” and “ $\{\textit{John}\} \rightarrow [\textit{sick}] \langle 1, 0.90 \rangle$,” the rule derives “ $\{\textit{John}\} \rightarrow \textit{patient} \langle 1, 0.45 \rangle$ ” (“John may be a patient”).

Therefore, induction and abduction can be seen as “inverse deduction,” in different ways [7], while inductive and abductive conclusions usually have lower confidence values than deductive conclusions, given the same truth-values of the premises.

The *revision* rule summarizes evidence from different sources for the same statement to get a more confident conclusion. It takes “ $S \rightarrow P \{w_1^+, w_1\}$ ” and “ $S \rightarrow P \{w_2^+, w_2\}$ ” as the premises, and derives “ $S \rightarrow P \{w_1^+ + w_2^+, w_1 + w_2\}$ ” as the conclusion. Here the truth-values are given as the amounts of evidence, which can be converted to and from the $\langle frequency, confidence \rangle$ pair. If from different bodies of evidence statement “ $\{John\} \rightarrow [sick]$ ” gets two different truth-values $\langle 1, 0.90 \rangle$ and $\langle 0, 0.80 \rangle$, respectively, then the revised conclusion has the truth-value $\langle 0.69, 0.93 \rangle$.

NAL contains other inference rules (and truth-value functions), which are described in [13] and other previous publications, but will be omitted here.

3 Testing Cases and Results

With the preparation of the previous sections, now we can describe our testing of NAL in the context of medical diagnosis. This domain is picked, both because of its great practical importance and its historical relation with AI applications. Since NAL is a *normative* model of reasoning, its design decisions cannot be directly evaluated and justified by comparing them with human psychological data, but by checking against the fundamental assumptions of the model [16]. Even so, it still makes sense to test such a system in a practical situation, to verify the applicability of its assumptions and the correctness of the implementation.

Especially, in this project we are interested in evaluating the expressive power of Narsese and the inferential power of NAL. That is, compared to the diagnosis process of a typical human doctor, whether all the domain knowledge can be expressed in Narsese, and whether all the usual inference steps can be formalized in NAL. Given this focus, the testing is done semi-automatically. In each step, the system is given a few Narsese sentences as premises and some derived conclusions are selected as the premises of the following steps, until a desired result is reached. In this way, the memory and control mechanism is excluded, so as to make the process simple and focused.

All the finished testing cases can be accessed from the first author’s website. In the following, a few sample cases are described in an edited version, to omit the implementation details.

Case 1. The system is given the following Narsese sentences as initial knowledge (the following English translations only roughly show their meaning):

- (1) $\{John-Doe\} \rightarrow [runny-nose]$ — “John Doe has a runny nose.”
- (2) $[runny-nose] \rightarrow [flu-symptom]$ — “Runny nose is a flu symptom.”
- (3) $flu-patient \rightarrow [flu-symptom]$ — “Flu patients have flu symptoms.”
- (4) $(\{Tamiflu\} \times flu-patient) \rightarrow treatment$ — “Tamiflu is a treatment for flu patients.”

As mentioned before, all declarations without an explicit truth-value will get the default value $\langle 1, 0.90 \rangle$ when accepted by the system.

From (2) and (1), the deduction rule derives

- (5) $\{John-Doe\} \rightarrow [flu-symptom]$ $\langle 1, 0.81 \rangle$ — “John Doe has flu symptoms.”

From (3) and (5), the abduction rule derives

(6) $\{John-Doe\} \rightarrow flu-patient \langle 1, 0.42 \rangle$ — “John Doe seems to be a flu patient.”

As mentioned before, (4) can be rewritten as

(7) $flu-patient \rightarrow (\perp treatment \{Tamiflu\} \diamond)$

From (7) and (6), the deduction rule derives

(8) $\{John-Doe\} \rightarrow (\perp treatment \{Tamiflu\} \diamond) \langle 1, 0.34 \rangle$ — “John Doe may get Tamiflu for treatment.”

In this way, the system reaches a diagnosis and a treatment suggestion, though the confidence values of the conclusions are quite low, caused by the use of non-deductive rules and the small amount of evidence.

Case 2. In addition to the initial knowledge of Case 1, assume the system also knows the following:

(9) $\{John-Doe\} \rightarrow [sore-throat]$

(10) $[sore-throat] \rightarrow [flu-symptom]$

Then from them, by deduction, the system can also gets

(11) $\{John-Doe\} \rightarrow [flu-symptom] \langle 1, 0.81 \rangle$

Though this conclusion looks identical to (5), it comes from a distinct body of evidence. Now the revision rule can take (5) and (11) as evidence, and gets

(12) $\{John-Doe\} \rightarrow [flu-symptom] \langle 1, 0.90 \rangle$

Using (12) in the place of (5) in Case 1, the following conclusions on diagnosis and treatment will both have higher confidence values than those obtained without the “sore throat” evidence. Similarly, adding more evidence for an intermediate conclusion will eventually increase the confidence of the final results.

Case 3. When a symptom appears in more than one type of sickness, the quantitative truth-value in the related knowledge will make a difference in the competing conclusions. For example, assume the system knows (1), “John-Doe has a running nose,” and the following information.

(13) $flu-patient \rightarrow [runny-nose] \langle 0.90, 0.95 \rangle$

(14) $cold-patient \rightarrow [runny-nose] \langle 1.00, 0.95 \rangle$

(15) $allergy-patient \rightarrow [runny-nose] \langle 1.00, 0.80 \rangle$

From them, the abduction rule derives, respectively,

(16) $\{John-Doe\} \rightarrow flu-patient \langle 1, 0.43 \rangle$

(17) $\{John-Doe\} \rightarrow cold-patient \langle 1, 0.46 \rangle$

(18) $\{John-Doe\} \rightarrow allergy-patient \langle 1, 0.42 \rangle$

Therefore, from the available evidence, “cold” is the most confident diagnosis.

Case 4. When a conclusion has negative evidence, its frequency will be decreased, though it will not be simply rejected as “false.” When the alternatives are even less likely, such a conclusion can still be considered as the most likely one. For example, if the system knows three symptoms of flu

(19) $flu-patient \rightarrow [runny-nose] \langle 0.90, 0.99 \rangle$

(20) $flu-patient \rightarrow [headache] \langle 0.75, 0.99 \rangle$

(21) $flu-patient \rightarrow [sore-throat] \langle 0.60, 0.99 \rangle$

while a patient only shows two of them

(22) $\{John-Doe\} \rightarrow [runny-nose]$

(23) $\{John-Doe\} \rightarrow [headache]$

(24) $\{John-Doe\} \rightarrow [sore-throat] \langle 0.00, 0.90 \rangle$

then the conclusion is

$$(25) \{John-Doe\} \rightarrow flu-patient \langle 0.73, 0.67 \rangle$$

Here the frequency value is more similar to a “degree of membership” as discussed in fuzzy logic [18, 11] than a probability value in the usual sense — John Doe is still judged to be a flu patient, though not a typical one.

Case 5. Even if there are lot of statistical data, a direct application of probability theory can still be difficult. For example, a patient may belong to two categories A and B, and they have different risks to have disease D [5]. In non-monotonic logic, a similar “multiple inheritance” problem is called the “Nixon Diamond” [8]. In NARS, the two competing conclusions can be merged by the revision rule [10]. When the two conclusions are equally strong, the conclusion does not favor either of the two. For example, from the following given knowledge,

$$(26) \{John-Doe\} \rightarrow [runny-nose]$$

$$(27) \{John-Doe\} \rightarrow [chest-pain]$$

$$(28) [runny-nose] \rightarrow flu-patient$$

$$(29) \text{“If someone has chest pain, then that person usually does not has flu.”}^2$$

the conclusion is

$$(30) \{John-Doe\} \rightarrow flu-patient \langle 0.50, 0.90 \rangle$$

However, when truth-values are explicitly specified, the premises usually do not provide exactly the same amount of positive and negative evidence for the conclusion, and in that case the system will have a non-trivial opinion, as in the following version of the same example:

$$(26') \{John-Doe\} \rightarrow [runny-nose] \langle 1.00, 0.99 \rangle$$

$$(27') \{John-Doe\} \rightarrow [chest-pain] \langle 1.00, 0.80 \rangle$$

$$(28') [runny-nose] \rightarrow flu-patient \langle 0.90, 0.99 \rangle$$

$$(29') \text{“If someone has chest pain, then that person usually does not has flu.”} \\ \langle 0.80, 0.99 \rangle$$

$$(30') \{John-Doe\} \rightarrow flu-patient \langle 0.81, 0.90 \rangle$$

Case 6. In NAL, all domain knowledge can be revised by new evidence, and the system can learn new beliefs, concepts, and skills. Actually, here “learning” and “reasoning” are carried out by the same underlying process, though the former word focuses on the long-term effect of the process, while the latter on the individual steps. For example, from

$$(31) \{John\} \rightarrow [runny-nose]$$

$$(32) \{John\} \rightarrow [fever]$$

NAL derives a conclusion

$$(33) \{John\} \rightarrow ([runny-nose] \cap [fever]) \langle 1.00, 0.81 \rangle$$

Here $([runny-nose] \cap [fever])$ may be a new concept that never existed in the system before. When a new concept is created, usually the system cannot fully determine its usefulness in summarizing experience. After all, from (31) and “ $\{John\} \rightarrow [back-pain]$,” the system can also create a new concept $([runny-nose] \cap [back-pain])$, which seems accidental. What the system does is

² The Narsese representation of this sentence uses *variable term*, *negated statement*, and *implication copula*, and therefore is beyond the scope of the previous NAL Overview. For how these items are formally defined, see [13].

to give each concept (either given or created) a priority value, which will then be adjusted according to the usefulness of the concept. In this way, concepts like $([runny-nose] \cap [fever])$ will gradually become part of the system's stable knowledge since it captures a repeatedly appearing pattern, while concepts like $([runny-nose] \cap [back-pain])$ will be gradually forgot. In this way, NAL not only specifies how the system learns and creates new *statements*, but also how it learns and creates new *terms* and *concepts*.

4 Comparison and Discussion

Since NAL has been compared to mathematical logic [10, 15], probability theory [12, 15], fuzzy logic [11], etc., in the *theoretical* assumptions and properties, in this paper we only compare the approaches from a *practical* point of view.

NAL works with *insufficient knowledge*, which has the following implications:

- Knowledge can be *uncertain*, and the uncertainty can be randomness, fuzziness, ignorance, and so on, or a mixture of them.
- Knowledge does not need to be *consistent*, as defined either in mathematical logic or in probability theory. The system can handle competing or conflicting conclusions by considering their evidential support.
- The evidence can arrive from time to time, and the system revises its beliefs according to the available evidence. In this sense, NAL is *non-monotonic*, though it is very different from the binary “non-monotonic logics.”
- NAL does not assume that a truth-value will eventually converge to an “objective” truth or a probability value.
- The system is *open* to knowledge of any content, as long as it can be expressed in Narsese. There is no “possible world” or “sample space” defined by a constant set of statements or terms.
- The system is called “non-axiomatic,” because there is no “axiom” among domain knowledge. The truth-value of a statement is always revisable by new evidence. All domain knowledge can be learned, rather than built into the system.
- Though more evidence is preferred, the system still produces a conclusion when the amount of evidence is less than desired, by making guesses and hypotheses, and marking their reliability with the confidence measurement.

To be used with *insufficient resources*, NAL has the following features:

- Each inference step is an *independent local* operation, meaning that only a few sentences are involved as premises and conclusions, and the step does not directly depend on the other knowledge or activities in the system. Therefore, such a step only requires a small constant amount of time and space.
- A complete problem-solving process (such as from patient symptoms to diagnosis and treatment) consists of a number of inference steps. However, since a problem usually can be solved in many different ways, in NAL the process does not follow a predetermined algorithm, but is handled in a case-by-case manner [14].

- The resource cost for a given problem is not a constant, but depends on the current context, that is, what relevant beliefs are there, in what order they are considered, how much resource the problem has obtained in the competition with co-existent problems and activities, etc. Therefore, NAL makes “anytime” responses [1].
- The system is *scalable* to large amounts of knowledge and complicated problems, not because these factors do not make the system’s algorithm intractable, but because NAL does not exhaustively search all possibilities when solving a problem. Instead, it only considers a “reasonable” amount of them, depending on the amount of available resources.

These properties are usually not possessed fully by reasoning systems based on traditional theories. However, it is not claimed that NAL is always superior than mathematical logic or probabilistic logic. Actually it is the opposite: whenever the knowledge/resource demands of a traditional model can be satisfied, it usually works better than NAL. It is when those demands cannot be satisfied, that NAL works better than illegally applying a traditional model, providing random responses, or simply giving up.

Compared to the traditional models of reasoning, NAL is closer to human reasoning. However, NAL is not designed as a descriptive model of human reasoning, so it does not have to fit the human data in all details. What is hoped is that it *follows the same principle* as human intelligence, so their working processes and capabilities should be similar to each other. Since human reasoning is not accurately defined, there is no way to exactly evaluate the similarities and difference between a formal model and human reasoning. What we can say about NAL is that in this testing project, we have not found any domain knowledge that cannot be expressed in Narsese, nor a common inference schema or pattern that cannot be captured by NAL’s rules.

5 Conclusions

Non-Axiomatic Logic (NAL) specifies valid inference steps under the assumption of insufficient knowledge and resources. Here “valid” means best supported by available evidence, that is, the evidence the system can collect under the existing resource restriction. Such a logic is fundamentally different from traditional logical models, where “valid” usually means “deriving absolute truth from absolute truth.”

Therefore, compared with other logical models, NAL makes weaker assumptions about what knowledge the system has and how much resources are affordable, and so can be applied to situations outside the applicable scope of the traditional models.

The recent testing in medical diagnosis shows that NAL can properly express the knowledge in that domain, as well as carry out the inference steps of a typical doctor. Though the system is not mature enough for actual applications yet, the potential of this technique is profound.

From an AGI point of view, the reasoning system approach has the advantage of clearly separating the domain-specific knowledge from the general-purpose logic and control mechanism. Furthermore, the notion of “reasoning” can be naturally extended to include some notions that are traditionally described as outside the scope of reasoning, such as learning, problem solving, and decision making. In this way, NAL provides a unified theory and model of intelligence.

References

1. Dean, T., Boddy, M.: An analysis of time-dependent planning. In: Proceedings of AAAI-88. pp. 49–54 (1988)
2. Donini, F.M., Lenzerini, M., Nardi, D., Schaerf, A.: Reasoning in description logics. In: Brewka, G. (ed.) *Principles of Knowledge Representation*, pp. 191–236. CSLI Publications, Stanford, California (1996)
3. Frege, G.: *Begriffsschrift, a formula language, modeled upon that of arithmetic, for pure thought*. In: van Heijenoort, J. (ed.) *Frege and Gödel: Two Fundamental Texts in Mathematical Logic*, pp. 1–82. iUniverse, Lincoln, Nebraska (1999), originally published in 1879.
4. Ginsberg, M. (ed.): *Readings in Nonmonotonic Reasoning*. Morgan Kaufmann, San Mateo (1987)
5. Kyburg, H.E.: The reference class. *Philosophy of Science* 50, 374–397 (1983)
6. Pearl, J.: *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann Publishers, San Mateo, California (1988)
7. Peirce, C.S.: *Collected Papers of Charles Sanders Peirce*, vol. 2. Harvard University Press, Cambridge, Massachusetts (1931)
8. Russell, S., Norvig, P.: *Artificial Intelligence: A Modern Approach*. Prentice Hall, Upper Saddle River, New Jersey, 3rd edn. (2010)
9. Wang, P.: *Non-Axiomatic Reasoning System: Exploring the Essence of Intelligence*. Ph.D. thesis, Indiana University (1995)
10. Wang, P.: Reference classes and multiple inheritances. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems* 3(1), 79–91 (1995)
11. Wang, P.: The interpretation of fuzziness. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 26(4), 321–326 (1996)
12. Wang, P.: Confidence as higher-order uncertainty. In: *Proceedings of the Second International Symposium on Imprecise Probabilities and Their Applications*. pp. 352–361. Ithaca, New York (2001)
13. Wang, P.: *Rigid Flexibility: The Logic of Intelligence*. Springer, Dordrecht (2006)
14. Wang, P.: Case-by-case problem solving. In: *Proceedings of the Second Conference on Artificial General Intelligence*. pp. 180–185 (2009)
15. Wang, P.: Formalization of evidence: A comparative study. *Journal of Artificial General Intelligence* 1, 25–53 (2009)
16. Wang, P.: The evaluation of AGI systems. In: *Proceedings of the Third Conference on Artificial General Intelligence*. pp. 164–169 (2010)
17. Whitehead, A.N., Russell, B.: *Principia Mathematica*. Cambridge University Press, Cambridge (1910)
18. Zadeh, L.A.: A theory of approximate reasoning. In: Hayes, J.E., Michie, D., Mikulich, L.I. (eds.) *Machine Intelligence*, vol. 9, pp. 149–194. Halstead Press, New York (1979)