Chapter 3

The Core Logic

The logic implemented in NARS is called *Non-Axiomatic Logic*, hereforth *NAL*. The formal language used in NARS is called *Narsese*.

In the following, both Narsese and NAL will be defined layer by layer, each of which adds new grammar rules and inference rules into the system. Overall, there will be eight layers, and the language used in NAL-n is Narsese-n (n = 1, ..., 8).

In this chapter, the simplest non-axiomatic logic, NAL-1, is defined. Narsese-1 is a simple language, in which each statement consists of two atomic terms, linked by an inheritance relation. NAL-1 has inference rules defined on this language.

To provide a proper semantics for this logic, an idealized version of the logic, NAL-0, is introduced first. This logic is not non-axiomatic, but will be used as part of the meta-language of NAL. Set theory and first-order predicate logic are also used as parts of the meta-language of NAL.

3.1 NAL-0: binary inheritance

NAL-0 is a simple binary deductive logic[1]. It is not actually "non-axiomatic," but we need it to define the semantics of NAL.

[1]It was formerly called Inheritance Logic in [Wang, 1994b, Wang, 1995a].

3.1.1 Language: term and inheritance

NAL-0, like all members of the NAL family, is a "term logic." This type of logic is different from predicate logics, because of its usage of *categorical* sentences and *syllogistic* inference rules [Bocheński, 1970, Copi, 1982, Englebretsen, 1996]. Therefore, it is also called "categorical logic" or "syllogistic logic."[2]

First, let us define the smallest unit of Narsese, "term."

Definition 1 *A* term*, in the simplest form, is a string of letters in an alphabet. In such a form, it is also called an* atomic term*.*

The default alphabet in this book is the alphabet of English plus digits 0 to 9, and most terms we use as examples are common English words, such as "bird", "animal", and "water" (we also allow hyphenated terms). It is easy for NAL to use words in another natural language (such as Chinese) as terms, because the following design does not depend on the choice of the alphabet or characters.

Later in this book, we will see that a term is the name of a concept in NARS. The above definition only gives the simplest form of terms, and more complicated forms will be introduced later.

Definition 2 *The* inheritance relation*, "$\rightarrow$", is a relation from one term to another term, and defined by* being reflexive and transitive*. An* inheritance statement *consists of two terms related by the inheritance relation. In the inheritance statement "$S \rightarrow P$", S is the* subject term *and P is the* predicate term*.*

In a statement, the two terms can be the same.

According to this definition, for any term X, "$X \rightarrow X$" is always true (reflexivity). Also, if "$X \rightarrow Y$" and "$Y \rightarrow Z$" are true, so is "$X \rightarrow Z$" (transitivity). On the other hand, if there is a relation defined among terms, which is both reflexive and transitive, and has no additional property, then it is the same as the inheritance relation defined in NAL.

[2] A detailed comparison between predicate logics and term logics will be given in Chapter 10.

The inheritance relation is neither symmetric nor anti-symmetric. That is, for different X and Y, given "$X \rightarrow Y$", whether "$Y \rightarrow X$" is also true cannot be determined.

NAL is a "term logic," partially because its sentences are "categorical," with a "subject-copula-predicate" format. The inheritance relation is a kind of "copula," and intuitively, it indicates that S is a *specialization* of P, and P is a *generalization* of S. It roughly corresponds to "S is a kind of P" in English. For example, "$apple \rightarrow fruit$" says that "Apple is a kind of fruit."

The inheritance relation defined above is closely related to many well-known relations, such as "belongs to" (in Aristotle's syllogisms), "subset" (in set theory), "IS-A" (in semantic networks) [Brachman, 1983], "inheritance assertion" (in inheritance systems [Touretzky, 1986]), and "inheritance" in object-oriented programming (such as "extends" in Java).

What makes the inheritance relation in NAL different from the other relations are:

1. It is a relation between two *terms* (not between sets, classes, or concepts).

2. The relation is defined completely (no more, no less) by the two properties, *reflexivity* and *transitivity*.

Now we use the above notions to define the corresponding version of Narsese.

Definition 3 Narsese-0 *is a formal language whose sentences are inheritance statements.*

Therefore, the grammar of Narsese-0 is given in Table 3.1.

3.1.2 Semantics: truth and meaning

Now let us establish a semantics for Narsese-0, by defining the notions of "truth" and "meaning" in the language.

For a reasoning system implementing the logic NAL-0, the language Narsese-0 is used for both internal knowledge representation and external communication. The initial knowledge of the system, obtained from the environment, is defined as its "experience."

$<\mathit{sentence}>$	::=	$<\mathit{statement}>$
$<\mathit{statement}>$	::=	$<\mathit{term}><\mathit{copula}><\mathit{term}>$
$<\mathit{copula}>$	::=	$\rightarrow$
$<\mathit{term}>$	::=	$<\mathit{word}>$
$<\mathit{word}>$	:	a string in a given alphabet

Table 3.1: The Grammar of Narsese-0

Definition 4 *The system's* experience*, K, is a non-empty and finite set of sentences in Narsese-0. In each statement in K, the subject term and the predicate term are different.*

For example, we can have $K = \{apple \rightarrow fruit,\ fruit \rightarrow plant\}$. As a set, K has no duplicated elements, and there is no order among its elements.

Definition 5 *Given experience K, the system's* beliefs*, K^*, is the transitive closure of K, excluding statements whose subject and predicate are the same term.*

Therefore, K^* is also a non-empty and finite set of sentences in *Narsese-0*, which includes K, as well as the sentences derived from K according to the transitivity of the inheritance relation. For the above K, $K^* = \{apple \rightarrow fruit,\ fruit \rightarrow plant,\ apple \rightarrow plant\}$. K^* can be generated from K in finite steps using an ordinary closure-generating algorithm.

In NARS, the words "belief" and "knowledge" are usually treated as exchangeable with each other.[3] Therefore, K^* can also be called the *knowledge base* of the system (as in some previous publications on NARS).

Now we can define the "truth value" of a statement and the "meaning" of a term, with respect to a given K.

Definition 6 *The* truth value *of a statement in NAL-0 is either* true *or* false*. Given experience K, the truth value of a statement is true if it is in K^*, or has the form of $T \rightarrow T$, otherwise it is false.*

[3] A justification of this decision will be given in Section 7.4.1.

Therefore we have two types of truth in NAL-0: *empirical* and *literal* (or call them *synthetic* and *analytic*, respectively). The former is "true according to experience," and the latter is "true by definition." Given the above definitions, truth in these two categories have no overlap.

In the following, we will call true statements "positive knowledge," and false statements "negative knowledge." All analytic truths are positive knowledge (and all of them fall into the pattern "$T \rightarrow T$"). Synthetic knowledge may be positive or negative. In NAL-0, negative knowledge are implicitly represented: they are not sentences in Narsese-0, but propositions in its meta-language. The amount of positive knowledge, (i.e., number of beliefs in K^*) increases monotonically with the increase of the experience K, but that is not the case for negative knowledge, which is implicitly defined by the former as "statements that not known to be true."

Therefore, here "true" is "derivable from experience," or "have the relation"; while "false" is "haven't found the relation," but not "have an anti-relation." NAL-0 accepts the "closed world" assumption, where "lack known relation" is treated as "no relation." It is necessary here, because if truth value is a summary of experience, then "having an unknown truth value" makes no sense.

For a term T that does not appear in K, all statements having T in them are false, except "$T \rightarrow T$." For example, given the above experience K, "$orange \rightarrow fruit$" is false. Though "$\neg(orange \rightarrow fruit)$" is not a valid sentence of Narsese-0, it is a valid proposition in the meta-language of NAL-0 (by taking statements of NAL-0 as propositions in propositional logic).

To clarify how a particular term T is related to other terms according to the experience of the system, the *extension* and *intension* of T are defined as the sets of its known specializations and generalizations, respectively:

Definition 7 *Given experience K, let the set of all terms appearing in K to be the* vocabulary *of the system, V_K. Then, the* extension *of a term T is the set of terms $T^E = \{x \mid x \in V_K \wedge x \rightarrow T\}$. The* intension *of T is the set of terms $T^I = \{x \mid x \in V_K \wedge T \rightarrow x\}$.*

Obviously, both T^E and T^I are determined with respect to K, so they can also be written as T^E_K and T^I_K. In the following, the simpler notions are used, with the experience K implicitly assumed.

Since "extension" and "intension" are defined in a symmetric way in NAL, for any result about one of them, there is a dual result about the other. Each belief of the system reveals part of the intension for the subject term and part of the extension for the predicate term. For example, "$apple \rightarrow fruit$" indicates that "*apple*" is in the extension of "$fruit$," and "$fruit$" is in the intension of "*apple*."

Theorem 1 *For any term* $T \in V_K$, $T \in (T^E \cap T^I)$*. If* T *is not in* V_K, $T^E = T^I = \{\}$*, though "*$T \rightarrow T$*" is still true.*

Since all the theorems in this book are easy to prove, I will leave the detailed proofs to the interested reader, and only explain their implications.

The above theorem states that any given term in V_K has a non-empty extension and a non-empty intension — both of them contain at least the term itself.

Definition 8 *Given experience* K*, the* meaning *of a term* T *consists of its extension and intension.*

Therefore, the meaning of a term is its relation with other terms, according to the experience of the system. A term T is "meaningless" to the system, if $T^E = T^I = \{\}$ (that is, it has never got into the experience of the system), otherwise it is "meaningful." The larger the extension and intension of a term are, the "richer" its meaning is.

Theorem 2 *If both* S *and* P *are in* V_K*, then* $(S \rightarrow P) \equiv (S^E \subseteq P^E) \equiv (P^I \subseteq S^I)$*.*

Here "$\equiv$" is the "if and only if" in propositional logic. This theorem says that "$S \rightarrow P$" is true if and only if the extension of S is fully contained in the extension of P, and also if and only if the intension of P is fully contained in the intension of S. In other words, the statement "There is an inheritance relation from S to P" is equivalent to both "P inherits the extension of S" and "S inherits the intension of P." This is the reason that "$\rightarrow$" is called an "inheritance" relation. Here I change the intuitive meaning of the word "inheritance" to indicate the situation where the two terms get different things from each other.

If "$S \to P$" is false, it means that the inheritance is incomplete — either $(S^E - P^E)$ or $(P^I - S^I)$ is not empty. However, it does not mean that S and P share no extension or intension.

Theorem 3 $(S^E = P^E) \equiv (S^I = P^I)$.

That is, the *extensions* of S and P precisely coincide if and only if their *intensions* precisely coincide. This means that in NAL-0 the extension and intension of a term are mutually determined. Consequently, one of the two uniquely determines the meaning of a term.

NAL-0 is a logic that from given experience determines the truth values of statements and meaning of terms, and this is what I call "experience-grounded semantics." I will come back to this topic again and again in this book.

3.1.3 Inference: rules and properties

As we can see from the above definitions, NAL-0 only has one inference rule, justified by the transitivity of the inheritance relation. As given in Table 3.2, this rule takes two statements as premises, and derive one statement as conclusion. This rule can be used to exhaust all beliefs according to a given experience.

$J_2 \setminus J_1$	$M \to P$	$P \to M$
$S \to M$	$S \to P$	
$M \to S$		$P \to S$

Table 3.2: The Inference Rule of NAL-0

There is also a "matching rule" in the system, which derives no new belief, but matches questions to answers.

Definition 9 *For different terms S and P, a* question *that can be answered with NAL-0 has one of the following three forms: (1) $S \to P?$,* *(2) $S \to ?$, and (3) $? \to P$. An empirical truth $S \to P$ is an answer to any of the three. If no such an answer can be found in K^*, "NO" is answered.*

The first form asks for an *evaluation* of a given statement, while the other two ask for a *selection* of a term with a given relation with another term.

If there are more than one answers to (2) and (3), any of them is equally good. Literal truth "$T \to T$" is a trivial answer to such a question, so it is not allowed.

The matching rule is shown in Table 3.3.

$J \setminus Q$	$S \to P?$	$S \to ?$	$? \to P$
$S \to P$	$S \to P$	$S \to P$	$S \to P$

Table 3.3: The Matching Rule of NAL-0

In NAL-0 the user cannot ask the system "What is not T?," because any term not appearing in K may become a (trivial) answer for this question.[4]

NAL-0 is consistent (since negative knowledge is implicitly represented), sound (since all derived statements are true), complete (since all truths are either literal, or in K^*), and decidable (since K^* can be generated in finite steps from a given K, and it can also be searched in finite time). Because in NAL-0 the time-space cost of inference is ignored, we do not need to worry about how the sentences are stored, and how the premises are chosen in each inference step.

NAL-0 is a term logic, with categorical statements, experience-grounded semantics, and syllogistic inference rules. However, it is not really a "non-axiomatic" logic, because it ignores the assumption of insufficient knowledge and resources. Though NAL-0 looks simple (and even trivial) by itself, its importance in defining the NAL family will be shown by the following sections.

[4]In NAL, there is a way to ask "What instances of S are not instances of P?," and it will be introduced in the next chapter.

3.2 The language of NAL-1

As mentioned in the previous chapter, a central issue in NARS is to treat "truth" as a matter of degree. For that purpose, we first define the concept of "evidence," then define "truth value" as a function of available evidence.

What is defined in Narsese-0 can be called "complete inheritance" (of extension/intension), and it can be naturally extended to the situation of "incomplete inheritance," and the concept of evidence will be introduced in the process.

3.2.1 Evidence and its measurement

As shown by a previous theorem, an inheritance statement is equivalent to a statement about the inclusion of extension (or intension) between two terms. Furthermore, such an inclusion can be seen as a summary of a set of inheritance statements. Based on this observation, "evidence" of an inheritance statement is defined as the following.

Definition 10 *For an inheritance statement "$S \rightarrow P$," its* evidence *are terms in S^E and P^I. Among them, terms in $(S^E \cap P^E)$ and $(P^I \cap S^I)$ are* positive evidence*, and terms in $(S^E - P^E)$ and $(P^I - S^I)$ are* negative evidence*.*

Here the related extensions and intensions are sets of terms, and "$\cap$" and "$-$" are the "intersection" and "difference" of sets, respectively, as defined in set theory.

Concretely, for a statement "$S \rightarrow P$" and a term M, if both "$M \rightarrow S$" and "$M \rightarrow P$" are true, it is positive evidence for the statement; if "$M \rightarrow S$" is true but "$M \rightarrow P$" is false, it is negative evidence. Symmetrically, if both "$P \rightarrow M$" and "$S \rightarrow M$" are true, it is positive evidence for the statement; if "$P \rightarrow M$" is true but "$S \rightarrow M$" is false, it is negative evidence.

Evidence is defined in this way, because as far as a term in positive evidence is concerned, the inheritance statement is correct; as far as a term in negative evidence is concerned, the inheritance statement is incorrect.

According to this definition, what counts as a piece of evidence is a *term*, not a *statement*. However, whether a given term M is positive or negative evidence for the statement "$S \to P$" is determined by two statements, one between M and S, and another between M and P.

Now we can rephrase the definition of truth value in NAL-0 in terms of "evidence": "$S \to P$" is true in NAL-0 if and only if according to the experience of the system, there is no negative evidence for the statement (that is, all available evidence is positive).

When a system has to answer questions with insufficient knowledge and resources, to only indicate whether there is (positive or negative) evidence is usually too rough as a summary of experience. When a statement has both positive and negative evidence, the system often needs to balance them, and takes into account the influence of future evidence. To do this, it is not enough to qualitatively indicate the existence of a certain type of evidence — we need to quantitatively *measure* evidence. For an adaptive system, though past experience is never sufficient to accurately predict future situations, the amount of evidence does matter for the system's decision, and the beliefs based on more evidence should be preferred.

Since according to the previous definition, terms in the extension or intension of a given term are equally weighted, the amount of evidence can be simply measured by the size of the corresponding set.

Definition 11 *For "$S \to P$," the amount of positive, negative, and total evidence is, respectively,*

$$
\begin{aligned}
w^+ &= |S^E \cap P^E| + |P^I \cap S^I| \\
w^- &= |S^E - P^E| + |P^I - S^I| \\
w &= w^+ + w^- \\
&= |S^E| + |P^I|
\end{aligned}
$$

For example, an observed black raven is a piece of positive evidence for "Raven is a kind of black-thing" ($w = w^+ = 1$), and an observed non-black raven is a piece of negative evidence for it ($w = w^- = 1$). Here we assume the observations have no uncertainty.

Amount of evidence captures the idea that an inheritance statement can be seen as a summary of some other inheritance statements.

An important feature of the above definition of evidence is that the "extensional factor" and the "intensional factor" are merged. From the amounts of evidence of a statement alone, there is no way to tell how much of it comes from extensional comparison or intensional comparison of the two terms. I will explain why this is desired later.

3.2.2 Truth value: frequency and confidence

Because all the operations in the system are based on available evidence, w^+ and w^- contain all the information about the uncertainty of the statement, as far as the current discussion is concerned. However, when represented in this way, the information is inconvenient for certain purposes, especially when we talk about beliefs where uncertainty is not obtained by directly counting evidence.

When comparing competing beliefs and deriving new conclusions, we usually prefer *relative measurements* to *absolute measurements*, because the evidence of a premise usually cannot be directly used as evidence for the conclusion. Also, it is often more convenient for the measurements to take values from a finite interval, while the amount of evidence has no upper bound. This point will become more clear later.

In principle, all intervals of real number can be mapped into the interval [0, 1], and this interval corresponds to notions like "ratio," "proportion," or "percentage," which are naturally used to represent "approximation" and "discount" in our daily life. Also, [0, 1] is a natural extension of the binary truth values, traditionally represented as $\{0, 1\}$. For these reasons, I use it for the uncertainty measurements in NARS.

A natural relative measurement for uncertainty is the *frequency*, or *proportion*, of positive evidence among all available evidence. In NAL, the "frequency" of a statement is defined as

$$f = w^+/w$$

If the system has observed 100 ravens, and 90 of them are black, but the other 10 are not, the system sets $f = 0.9$ for "Raven is a kind of black thing." When $w = 0$ (and therefore $w^+ = 0$), f is defined to be 0.5.

Although f is a natural and useful measurement, it is not enough for our current purpose. Intuitively, we have the feeling that the uncertainty evaluation $f = 0.9$ is uncertain itself. For a simple example, let us consider the following two situations: (1) the system only knows 10 ravens, and 9 of them are black, and (2) the system knows 10000 ravens, and 9000 of them are black. Though in both situations we have $f = 0.9$, the first case is obviously "more uncertain" than the second. Because here the uncertainty is about the statement "The frequency for ravens to be black is 0.9," we are facing a *higher-order* uncertainty, which is the uncertainty of an evaluation about uncertainty.

As mentioned previously, in NARS the uncertainty in a statement appears as the result of insufficient knowledge. Specially, the first-order uncertainty, measured by frequency, is caused by *known* negative evidence, and the higher-order uncertainty is caused by *potential* negative evidence. For the second measurement, we are looking for a function of w, call it c for *confidence*, that satisfies the following conditions:

1. Confidence c is a continuous and monotonically increasing function of w. (More evidence, higher confidence.)

2. When $w = 0$, $c = 0$. (Without any evidence, confidence is minimum.)

3. When w goes to infinity, c converges to 1. (With infinite evidence, confidence is maximum.)

There are infinite functions satisfying the above requirements, therefore we need more intuition to pick up a specific one.

Many functions with value range [0, 1] can be naturally interpreted as a *proportion* of a certain amount in a total amount. Following this path, when comparing available evidence to potential evidence, we might want to define c as the ratio of "the amount of evidence the system has obtained" to "the amount of evidence the system will obtain." Obviously, the first item is w, but for a system that is always open to new evidence, the second item is infinity, therefore the ratio is always 0. When compared with an infinite "future," the difference among the various finite "past" cannot be perceived. Therefore, it makes little sense to talk about an infinite future.

However, it makes perfect sense to talk about the *near* future. What the system needs to know, from the value of w, is how *sensitive* a frequency will be to new evidence; then the system can use this information to make a choice among competing beliefs. If we limit our attention to a future of fixed horizon, we can represent the information in w in a *ratio* form.

Let us introduce a positive number k, whose value can be metaphorically thought of as the distance to the (temporal) horizon, in the sense that k is the number of times we will still test the given inheritance statement. With this new notion of "horizon," measured by k, we can define a new measurement — confidence, in terms of the amount of total evidence w.

Now we get the relation between c and w in NAL:

$$c = w/(w + k)$$

where k is a positive parameter indicating the evidence to be collected in the "near future." Obviously, this function satisfies the three requirements listed previously.

In this way, the frequency and confidence of a statement are independent of each other, in the sense that, from the value of one, the other's value cannot be determined, or even estimated or bounded (except the trivial case where $c = 0$ implies $f = 0.5$).

For a specific system, k should remain fixed to make the system's behaviors consistent, but different systems can have different values for k. In this book, the default value of k is 1 (and we will discuss the choice of k later). Under such a definition, confidence indicates the ratio of the current amount of evidence to the amount of evidence the system will have after it gets new evidence with a unit amount. The more the system already knows about a statement, the less the new evidence will contribute (relatively), therefore the more confident, or the less ignorant, the system is, on the given statement. When $w = 1$, $c = 0.5$, and the new evidence will double the amount of available evidence; When $w = 999$, $c = 0.999$, and the new evidence will have little effect on the system's belief.

Together, f and c form the truth value of a statement in NAL, and they are defined by the amount of evidence.

Definition 12 *The truth value of a statement consists of a pair of real numbers in [0, 1]. One of the two is called* frequency*, computed as* $f = w^+/w$ *(or 0.5 if* $w = 0$*); the other is called* confidence*, computed as* $c = w/(w + k)$*, where* k *is a positive number.*

From a given truth value, the amount of positive, negative, and total evidence can be uniquely determined. Therefore, the "truth value" representation of uncertainty is functionally equivalent to the "amount of evidence" representation.[5]

3.2.3 Frequency interval

Interestingly, there is a third way to represent the uncertainty of a statement in NAL: as an interval of the frequency of success.

Given the above definition of frequency, after the coming of evidence of the amount k, the new f value will be in the interval

$$[w^+/(w + k),\ (w^+ + k)/(w + k)]$$

This is because the current frequency is w^+/w, so in the "best" case, when all evidence in the near future is positive, the new frequency will be $(w^+ + k)/(w + k)$; in the "worst" case, when all evidence in the near future is negative, the new frequency will be $w^+/(w + k)$.

Let us define this interval formally.

Definition 13 *The* lower frequency *of a statement,* l*, is* $w^+/(w + k)$*; the* upper frequency *of a statement,* u*, is* $(w^+ + k)/(w + k)$*. The* frequency interval *of the statement is* $[l, u]$*.*

This measurement has certain intuitive aspects in common with other interval-based approaches [Bonissone, 1987, Kyburg, 1988]. For example, the *ignorance* about where the frequency will be (in the near future) can be represented by the *width* of the interval, $i = u - l$. In NAL, i happens to be $1 - c$, so *ignorance* and *confidence* are complementary to each other.

It is important to remember that in NAL the interval $[l, u]$ indicates the range in which the frequency will lie in the *near* future, rather than

[5] The relations between this representation and other representations of uncertainty, such as probability and fuzziness, will be discussed in detail in Chapter 8.

in the *remote* future beyond that. According to the definition of truth value, with the coming of new evidence for a given statement, its confidence value monotonically increases, and eventually converges to 1, but its frequency may increase or decrease, and does not necessarily converge at all. For this reason, the frequency interval cannot be interpreted as indicating where the frequency will eventually be.

The interval representation of uncertainty provides a mapping between the "accurate representation" and the "inaccurate representation" of uncertainty, because "inaccuracy" corresponds to willingness to change a value within a certain range.

Within the system, it is necessary to keep an accurate representation of the uncertainty for statements, but it is often unnecessary for communication purposes. To simplify communication, uncertainty is often represented by a verbal label. In this situation, the truth value corresponds to the relative ranking of the label in the label set.

If in a language there are only N words that can be used to specify the uncertainty of a statement, and all numerical values are equally possible, the most informative way to communicate is to evenly divide the [0, 1] interval into N section: [0, 1/N], [1/N, 2/N], ..., [(N-1)/N, 1], and use a label for each section.

For example, if the system has to use a language where "false," "ambivalent" and "true" are the only valid words to specify truth value, and it is allowed to say "I don't know," then the most reasonable approach for input is to map the three words into [0, 1/3], [1/3, 2/3], and [2/3, 1], respectively, and ignore all "I don't know." For output, all conclusions whose confidence is lower than 1/3 become "I don't know," and for the others, one of the three words is used, according to the section in which the frequency of the conclusion falls.

A special situation of this is to use a single number, with its accuracy, to carry out both frequency and confidence information. In such a situation, "The frequency of statement S is 0.9" is different from "The frequency of statement S is 0.900" — though both give the same frequency, they give different confidence value. In the former case, the interval is [0.85, 0.95], so the confidence is $1 - (0.95 - 0.85) = 0.9$. In the latter case, the interval is [0.8995, 0.9005], so the confidence is $1 - (0.9005 - 0.8995) = 0.999$.

With the interval representation of uncertainty, NARS gains some flexibility in its communication. Though within the system, every belief is attached with numerical uncertainty measurement, in communications it is not necessary when accuracy is not required.

3.2.4 Relations among representations of uncertainty

Now we have three functionally equivalent ways to represent the uncertainty of a statement:

1. as a pair of *amounts of evidence* $\{w^+, w\}$, where $0 \leq w^+ \leq w$ (they do not have to be integers);

2. as a truth-value $<f, c>$, where both f and c are real numbers in $[0, 1]$, independent of each other;

3. as a *frequency interval* $[l, u]$, where $0 \leq l \leq u \leq 1$.

To avoid confusion, three types of brackets (“{},” “<>,” and “[ ]”) are used in this book for the three forms of uncertainty, respectively. Again, each of them is calculated with respect to certain part of the system's experience, which is implicitly assumed.

Formulas for inter-conversion among the three truth-value forms are displayed in Table 3.4.

This table can be easily extended to include w^- (the amount of negative evidence) and i (degree of ignorance). In fact, any valid (not inconsistent or redundant) assignments to any two of the eight measurements (for example, setting $w^+ = 3.5$ and $i = 0.1$, or setting $f = 0.4$ and $l = 0.3$) will uniquely determine the values of all the others. Therefore, the three forms of uncertainty measurement can even be used in a mixed manner.

Having several closely related forms and interpretations for uncertainty has the following advantages:

1. It gives us a better understanding of what uncertainty of statement really means in NARS, since we can explain it in different ways. The mappings also give us interesting relations among the various uncertainty measurements.

to \ from	$\{w^+, w\}$	$<f, c>$	$[l, u]$
$\{w^+, w\}$		$w^+ = k\frac{fc}{1-c}$ $w = k\frac{c}{1-c}$	$w^+ = k\frac{l}{u-l}$ $w = k\frac{1-(u-l)}{u-l}$
$< f, c >$	$f = \frac{w^+}{w}$ $c = \frac{w}{w+k}$		$f = \frac{l}{1-(u-l)}$ $c = 1 - (u - l)$
$[l, u]$	$l = \frac{w^+}{w+k}$ $u = \frac{w^+ + k}{w+k}$	$l = fc$ $u = 1 - c(1 - f)$	

Table 3.4: The Relations Among Forms of Truth-Value

2. It provides a user-friendly interface. If the environment of the system consists of human users, the uncertainty of a statement can be expressed in different ways, such as, "I've tested it w times, and in w^+ of them it was true," or "Its past success frequency was f, and the confidence was c," or "I'm sure that its success frequency will remain in the interval $[l, u]$ in the near future." We can maintain a single form as the internal representation (in the current implementation, it is the truth-value form), and, using the mappings in the above table, translate it into/from the others in the interface of the system when necessary.

3. It makes the designing of inference rules easier. For each rule, there should be a function that calculates the truth value of the conclusion from the truth values of the premises, with different rules of course equipped with different functions. As we will see in the following, for some rules it is easier to choose a function if we directly deal with truth values, while for other rules we may prefer to convert truth values into amounts of evidence, or frequency intervals.

4. It facilitates the comparison between measurements in NARS and the uncertainty measurements of various other approaches, because different forms capture different intuitions about uncertainty.[6]

[6] These comparisons will be left to Chapter 8.

Given experience K (as a finite set of binary inheritance statements), for an inheritance relation "$S \rightarrow P$" derived from it, w is always finite. Also, since the system has no need to keep statements for which there is no evidence, w should be larger than 0. For uncertainty represented in the other two forms, these translate into $0 < c < 1$ and $l < u$, $u - l < 1$, respectively.

Beyond the above normal values of uncertainty, there are two limit cases useful for the interpretation of uncertainty and the design of inference rules:

Null evidence: This is represented by $w = 0$, or $c = 0$, or $u - l = 1$, and of course means that the system knows nothing at all about the statement.

Full evidence: This is represented by $w = \infty$, or $c = 1$, or $l = u$. It means that the system already knows everything about the statement — no future modification of the uncertainty value is possible.

Though the above values never appear in actual beliefs of the system, they play important role in system design.

3.2.5 Narsese-1 and experience

Now let me summarize the grammar of Narsese-1 in Table 3.5. We can see that it is similar to that of Narsese-0, except that a binary "statement" plus its truth value becomes a multi-valued "judgment."

In the interface of the system, the other two types of uncertainty representation can also be used in place of the truth value of a judgment, though within the system they will be translated to (from) truth value. Also, truth values corresponding to "null evidence" and "full evidence" are not allowed to appear in the interface (or within the system), though they are used in the meta-language, as limit points, when the inference rules are determined.

Now we can treat Narsese-0 (defined in Table 3.1) as a subset of Narsese-1. In Narsese-1, "$S \rightarrow P <1, 1>$" indicates that the inheritance is complete (and negative evidence can be practically ignored), so it is identical to "$S \rightarrow P$" in Narsese-0.

$<\mathit{sentence}>$	::=	$<\mathit{judgment}> \mid <\mathit{question}>$
$<\mathit{judgment}>$	::=	$<\mathit{statement}><\mathit{truth\text{-}value}>$
$<\mathit{question}>$	::=	$<\mathit{statement}>?$
		$\mid\ ?<\mathit{copula}><\mathit{term}> \mid <\mathit{term}><\mathit{copula}>?$
$<\mathit{statement}>$	::=	$<\mathit{term}><\mathit{copula}><\mathit{term}>$
$<\mathit{copula}>$	::=	$\rightarrow$
$<\mathit{term}>$	::=	$<\mathit{word}>$
$<\mathit{truth\text{-}value}>$	:	a pair of real number in$[0,1] \times [0,1]$
$<\mathit{word}>$	:	a string in a given alphabet

Table 3.5: The Grammar of Narsese-1

In this way, the semantics of Narsese-1 is defined by a subset of the language, Narsese-0. Given the experience of the system K in Narsese-0, the binary inheritance language, the truth value of a judgment in Narsese-1, with subject and predicate in V_K, can be determined by comparing the meaning of the two terms. All these judgments form the beliefs of the system, K^*.

Similarly, we extend the concept of "meaning." For a system whose beliefs are represented in Narsese-1, the meaning of a term still consists of the term's extensional and intensional relations with other terms, as in NAL-0. The only difference is that the definition of extension and intension is modified as follows:

Definition 14 *A judgment "$S \rightarrow P <f, c>$" states that S is in the extension of P and that P is in the intension of S, with the truth value of the judgment specifying their degrees of membership.*

Consequently, extensions and intensions in NAL-1 are no longer ordinary sets with well-defined boundaries (as in NAL-0). They are similar to fuzzy sets [Zadeh, 1965], because terms belong to them to different degrees. What makes them different from fuzzy sets is how the "membership" is measured (in NAL, two numbers are used) and interpreted (in NAL, it is experience-grounded).[7]

[7]This issue will be discussed in detail in Section 8.2.

Given any set of statements of Narsese-0 as the experience of a NAL-1 system, the truth values of judgments and the meanings of terms can be determined. In this way, Narsese-0 is used as a meta-language of Narsese-1. At the same time, the former is a subset of the latter. Therefore, the experience-grounded semantics for Narsese is established in a "bootstrapping" manner.

However, since NAL-1 is used with insufficient knowledge and resources, its *actual* experience is a stream of sentences in Narsese-1, not a set of statements in Narsese-0 (as the idealized experience used in the above semantics).

There are several important differences between the "idealized experience" and the "actual experience" of the system.

- A judgment in the idealized experience has truth value $<1, 1>$, while a judgment in the actual experience has truth value $<f, c>$, where c is less than 1.

- The idealized experience is a set of sentences, which is available altogether to the system at the beginning, while the actual experience is a stream of sentences, coming to the system one at a time.

- For a given statement, in the idealized experience each piece of evidence is equally weighted, while in the actual experience, pieces of evidence make different contributions to its truth value, depending on several factors (to be described later).

- When *defining* the truth value of a judgment, the whole idealized experience is considered, while when *calculating* the truth value of a judgment, an inference rule only takes part of the actual experience into account.

- The idealized experience is used at design time to define truth value (of statements) and meaning (of terms), as well as to justify the inference rules, while the actual experience is used at run time by the inference rules to derive new statements (or terms) with their truth value (or meaning), or to modify the existing ones.

For example, if the system has a belief "$S \rightarrow P$ $<0.75, 0.80>$," then from the relationship between the truth value and the amount of evidence (and assuming $k = 1$), we get $w = 4$, $w^+ = 3$. Therefore, the system believes the statement "$S \rightarrow P$" to such an extent, *as if* it had tested the statement 4 times in idealized situations (by checking common elements of the extensions or the intensions of the two terms), in which the relation had been confirmed 3 times, and disproved 1 time. This does not imply, of course, that the system actually got the truth value by carrying out such tests — such absolute certainty can never be obtained in real life. Indeed, the system may have checked the relation more than four times in less-than-ideal situations (i.e., with results represented by judgments whose confidence values are less than 1), or the conclusion may have been derived from other beliefs, or even directly provided by the environment. But no matter how the truth value $< 0.75, 0.80 >$ is generated in practice (there are infinitely many ways it could arise), it can always be *understood* in a unique way, as stated above.

For any approach to extend a binary logic to a multi-valued logic, there is always the question for the meaning of the numerical truth value that need to be answered to make everything else meaningful, while "numerical statements are meaningful insofar as they can be translated, using the mapping conventions, into statements about the original qualitative structure" [Krantz, 1991]. In other words, "ideal experience" is being used in NAL as an "ideal meter-stick" to measure degrees of certainty. Like all measurements, though its unit is *defined* in an idealized situation, it is not *used* only in idealized situations — when we say that a cord is "3 meters long," we do not mean that we have compared it with three end-to-end meter-sticks.

Clearly, the actual experience of NARS is much more complex than the ideal experience as defined in the semantics, but it does not prevent us from saying that the truth value of a judgment summarizes its evidential support, and that the meaning of a term is derived from its experienced relations with other terms.

3.3 The inference rules of NAL-1

Now we can define inference rules for NAL-1, whose premises and conclusions are judgments of Narsese-1, and whose validity is justified according to the experience-grounded semantics.

3.3.1 Revision rule

In NAL, *revision* indicates the inference step in which evidence from different sources is combined. For example, assuming the system's previous uncertainty for "Ravens are black" is $<9/10,\ 10/11>$ (we know that it corresponds to "10 ravens observed, and 9 of them are black" when $k = 1$), now a new judgment comes, which is "Ravens are black $<3/4,\ 4/5>$" (so it corresponds to "4 ravens are observed, and 3 of them are black"). If the system can determine that no evidence is repeatedly counted in the two sources, then the uncertainty of the revised judgment should be $<6/7,\ 14/15>$ (corresponding to "14 ravens observed, and 12 of them are black").

Formally, the revision rule is defined in Table 3.6, where S can be any statement. The two premises may be conflicting to each other (when the two frequency values are very different), though this is not necessarily the case. Conflicting or not, the information in the two should be summarized into the conclusion.

$J_2 \setminus J_1$	$S < f_1,\ c_1 >$
$S < f_2,\ c_2 >$	$S < F_{rev} >$

Table 3.6: The Revision Rule

Since in this case the evidence of either premise is also evidence for the conclusion, and there is no overlapping evidence between the two premises, we have

$$w^+ = w_1^+ + w_2^+,\ w = w_1 + w_2$$

Then, according to the relationship between truth value and amount of evidence, we get the truth-value function for the revision rule:

$$F_{rev}:\ f = \frac{f_1c_1(1-c_2)+f_2c_2(1-c_1)}{c_1(1-c_2)+c_2(1-c_1)},\quad c = \frac{c_1(1-c_2)+c_2(1-c_1)}{c_1(1-c_2)+c_2(1-c_1)+(1-c_1)(1-c_2)}$$

This function has the following properties:

- The order of the premises does not matter.

- As a weighted average of f_1 and f_2, f is usually a "compromise" of them, and is closer to the one that is supported by more evidence.
- The value of c is never smaller than either c_1 or c_2, that is, the conclusion is supported by no less evidence than either premise.
- If $c_1 = 0$ and $c_2 > 0$, then $f = f_2$ and $c = c_2$, that is, a judgment supported by null evidence cannot revise another judgment.
- If $c_1 = 1$ and $c_2 < 1$, then $f = f_1$ and $c = c_1$, that is, a judgment supported by full evidence cannot be modified by empirical evidence.

Because actual confidence values are always in (0, 1), the last two cases do not actually appear at run time, but serve as limit situations. Also because of this reason, it does not matter for the above function has undefined value when $c_1 = c_2 = 0$ and $c_1 = c_2 = 1$.

This definition is compatible with our intuition about evidence and revision — revision is nothing but to reevaluate the uncertainty of a statement by taking new evidence into account. Revision is not updating, where old evidence is thrown away.[8] A high w means that the system already has much evidence for the statement, therefore its confidence is high and its ignorance is low, and consequently the judgment is relatively insensitive to new evidence. All these properties are independent to the decisions on how w is divided into w^+ and w^-, as well as to how they are actually measured (so these decisions may change from situation to situation without invalidating the revision rule).

What happens in revision is similar to what Keynes said: "As the relevant evidence at our disposal increases, the magnitude of the probability of the argument may either decrease or increase, according as the new knowledge strengthens the unfavorable or the favorable evidence; but *something* seems to have increased in either case — we have a more substantial basis upon which to rest our conclusion." [Keynes, 1921].

It needs to be clarified that here "revision" refers to the operation by which the system summarize two (maybe conflicting) beliefs.

[8] This issue will be discussed with more details in Section 8.3.1.

In this operation the conclusion always has a higher confidence. However, generally speaking, in NARS it is possible for the system to lose its confidence in a belief. This can be caused by the "forgetting" or "explaining away" of previously available evidence. This issue will be discussed later, after other relevant components of NARS are introduced.

Now the remaining issue in revision is how to recognize and handle the "overlapping evidence" situation. For that, we need to record, for each judgment, the fragments of experience its truth value is based on.

Definition 15 *If J is an input judgment that appears in the system's experience, with a unique serial number N, it is based on the fragment of experience $\{N\}$. If J is derived from premises $J_1, \cdots, J_n$, which are based on the fragments of experience $K_1, \cdots, K_n$, respectively, then J is based on fragment $K_1 \cup \cdots \cup K_n$.*

The serial numbers will be generated by the program that implements NAL. If the same judgment appears twice in the system's experience, each occurrence will have its own serial number, and the two occurrences will later be treated as different pieces of evidence. On the contrary, if one occurrence produces multiple copies in the system, they will all have the same serial number, and be treated as the same piece of evidence.

The system is designed in this way, because for an adaptive system, what really matters is to predict whether a given statement will be true next time. For someone who lives on a small island with a black swan, "Swan is black" should have a higher frequency than "Swan is white" — though the person has the knowledge that most swans in the world are white, "black swan" appears more often in his/her personal experience. Of course, we do not want to count the same observation more than once, but different observations of the same swan should be treated as multiple pieces of evidence. Therefore, accurately speaking, in NARS the truth value attached to "Swan is black" is not about *how many swans (in the world) are black*, but about *how often a black swan (in the system's experience) is encountered.*

If judgments J_1 and J_2 are based on fragments of experience K_1 and K_2, respectively, and K_1 and K_2, as sets of serial numbers, have no common elements, then the evidence supporting the two judgments do not overlap with each other (that is, no piece of evidence is used to calculate the truth values of both premises). If the two judgments are about the same statement, then they can be used by the revision rule as premises to derive a (summarized) conclusion.

With insufficient resources, NARS cannot maintain a complete record of the supporting experience for each judgment, because it may ask for time and space that the system cannot afford. Therefore the "overlapping-evidence recognition problem" cannot be completely solved by a system with insufficient resources.

Obviously, this limitation holds also for human beings: we could not possibly remember all evidence that supports each judgment we make. Nevertheless, NARS needs to be able to handle this problem somehow, which is not limited to revision only; otherwise, as Pearl points out, "a cycle would be created where any slight evidence in favor of A would be amplified via B and fed back to A, quickly turning into a stronger confirmation (of A and B), with no apparent factual justification." [Pearl, 1988].

The NARS strategy for dealing with this problem is to record only a *constant-sized fragment* of the experience supporting each judgment, and to use such fragments to determine approximately whether two judgments are based on overlapping evidence. As mentioned above, each input judgment is automatically assigned a unique serial number when accepted by the system. In each inference step, the conclusion is assigned a list of serial numbers constructed by interleaving its parents' (the premises') serial-number lists, and then truncating that list at a certain length.

For example, suppose the maximum length for serial-number lists is 4. In this case, if two judgments have a parent or grandparent judgment in common, their serial-number lists will overlap. Now the revision rule is applied only if the two premises' serial-number lists have no common elements, meaning that they are related, if at all, more than two "generations" ago. This mechanism is only an approximation to the perfect solution to the problem, of course.

Though not perfect, it is a reasonable solution when resources are insufficient, and "reasonable solutions" are exactly what we expect from a non-axiomatic system. It is also similar to the strategy of the human mind, since we usually have impressions about where our judgments come from, but such impressions are far from complete and accurate. Also, there is no guarantee that we never repeatedly using the same evidence to adjust our degree of belief.

3.3.2 Choice rule

What should NARS do when two conflicting judgments $S < f_1, c_1 >$ and $S < f_2, c_2 >$ are based on overlapping evidence?

Ideally, we would like to record the precise contribution of each input judgment, and then to subtract the amount of the overlapping evidence from the truth value of the conclusion, so that nothing is counted more than once. Unfortunately, this is impossible, because the experience recorded for each judgment is incomplete, as has just been explained. Furthermore, to find out the contribution of a given input judgment to the overall conclusion is very difficult, and simply impossible given incomplete records.

Nevertheless, NARS needs to be able to handle this situation. For example, the two conflicting judgments may be candidate answers to an evaluative question. If it is impossible to combine them, then NARS needs to make a choice between the two. In the current situation, the *choice* rule is very simple: the judgment having a *higher confidence* (no matter what its *frequency* is) is taken as the better answer, the idea being that if an *adaptive* system must make a choice between conflicting judgments, the one based on more experience has higher priority.

To make a choice between two competing answers for a selective question is more complicated. Let us say that the system is asked the selective question "$S \rightarrow$?," meaning that it should come up with a term T that is a "typical element" in the intension of S (not S itself, of course). Ideally, the best answer would be provided by a judgment "$S \rightarrow T < 1, 1 >$." But of course this is impossible, because confidence can never reach 1 in NARS. Therefore, we have to settle for the best answer the system can find under the constraints of available knowledge and resources.

Suppose the competing answers are "$S \rightarrow T_1 < f_1, c_1 >$" and "$S \rightarrow T_2 < f_2, c_2 >$." Which one would be better? Let us consider some special cases first:

1. When $c_1 = c_2$, the two answers are supported by the same amount of evidence. For example, both come from statistical data of 100 samples. Obviously, the answer with the *higher frequency* is preferred, since that statement has more positive evidence than the other.

2. When $f_1 = f_2 = 1$, all available evidence is positive. Now the answer with the *higher confidence* is preferred, since it is more strongly confirmed by experience.

3. When $f_1 = f_2 = 0$, all available evidence is negative. Now the answer with the *lower confidence* is preferred, since it is less strongly refuted by the experience. Of course such an answer is still a bad one because of its negative nature, but it may be the best (the least negative) answer the system can find for the question.

From these special cases, we can see that to set up a *general rule* to make a choice among competing judgments, we need somehow to combine the two numbers in a truth value into a single measurement. The current situation is different from the previous one. "$S \rightarrow T_1 < f_1, c_1 >$" and "$S \rightarrow T_2 < f_2, c_2 >$" do not conflict with each other — they have different contents — but they compete for being the "best supported intensional relation of S."

In NARS, an *expectation* measurement, e, is defined on every judgment for this purpose. Different from truth value (which is used to record past experience), expectation is used to predict future experience. "$e = 1$" means that the system is absolutely sure that the statement will always be confirmed by future experience; "$e = 0$" means it will always be refuted; and "$e = 0.5$" means the system considers it equally likely to encounter a piece of positive or a negative evidence.

To calculate e from $< f, c >$, we can see that under the assumption that the system makes extrapolations from its (past) experience, it would be natural to use f as e's "first-order approximation." However, such a *maximum-likelihood estimate* is not good enough when c is small [Good, 1965]. For example, if a hypothesis has been tested only once, it would not make sense to set one's expectation to 1 (if the test was a success) or to 0 (if the test was a failure).

Intuitively, e should be more "conservative" (i.e., closer to 0.5, the "no-preference point") than f, to reflect the fact that the future may be different from the past. Here is where the confidence c affects e — the more evidence the system has accumulated, the more confident the system is (indicated by a larger c) that its predicted frequency e should be close to its experienced frequency f. Therefore, it is natural to define

$$F_{exp}: \ e = c(f - 0.5) + 0.5.$$

In particular, when $c = 1$ (full evidence), $e = f$; when $c = 0$ (null evidence), $e = 0.5$. Alternatively, this equation can be rewritten as $c = (e - 0.5)/(f - 0.5)$ (when $f \neq 0.5$), showing that c indicates the ratio of e's and f's distances to 0.5.

To express the definition of e in the other two forms of uncertainty leads to interesting results.

When the uncertainty is represented as a frequency interval, from the inter-conversion formulas in Table 3.4, we get

$$e = (l + u)/2$$

Thus e is precisely the *expectation of the future frequency* — that is, the midpoint of the interval in which the frequency will lie, in the near future.

When the uncertainty is represented as amounts of evidence, from the mappings in Table 3.4 we get

$$e = (w^+ + k/2)/(w + k)$$

which is a continuum (i.e., a family) of functions with k as a parameter. This formula turns out to be closely related to what has been called the "beta-form based continuum" (with positive and negative evidence weighted equally) [Good, 1965], and the "λ-continuum" (with the "logical factor," or prior probability, being 1/2) [Carnap, 1952]. Though interpreted differently, the three continuum share the same formula and make identical predictions. All three continua have *Laplace's law of succession* as a special case (when $k = 2$), where the probability of success on the next trial is estimated by the formula $(w^+ + 1)/(w + 2)$.

Now we can see how the choice of the parameter k can influence the behavior of a system. Let us compare a system A_1 with $k = 1$ and a system A_2 with $k = 10$. The problem is to make a choice between two competing answers "$S \rightarrow P_1 \ \{w_1^+, w_1\}$" and "$S \rightarrow P_2 \ \{w_2^+, w_2\}$" (where the truth values are represented as weights of evidence). It is easy to see that when $w_1 = w_2$ or $w_1^+/w_1 = w_2^+/w_2$, the two systems make the same choice. It is only when a system needs to make a choice between a higher f and a higher c that the value of k will matter. For example, let us suppose that $w_1^+ = w_1 = 2, w_2^+ = 5$, and $w_2 = 6$. In this situation, in A_1, $e_1 = (2+0.5)/(2+1) \approx 0.83$, $e_2 = (5+0.5)/(6+1) \approx 0.79$, and thus A_1 will choose the first answer (since all of its evidence is positive); in A_2, $e_1 = (2+5)/(2+10) \approx 0.58$, $e_2 = (5+5)/(6+10) \approx 0.63$, and thus A_2 will choose the second answer (since it is more fully tested, and its frequency is not much lower than that of the other alternative).

Therefore, k is one of the "personality parameters" of the system, in the sense that it indicates a certain systematic preference or bias, for which there is no "optimal value" in general. The larger k is, the more "conservative" the system is, in the sense that the system always makes smaller adjustments when e is reevaluated according to new evidence, than a system having a smaller value of k. This parameter was called the "flattening constant" by Good ([Good, 1965], where he also tried to estimate its value according to certain factors that are beyond our current consideration), and was interpreted by him as a way to choose a prior probability distribution. The same parameter was interpreted by Carnap as the "relative weight" of the "logical factor" [Carnap, 1952].

In summary, the choice rule is formally defined in Table 3.7, where $S_1 \ {<} f_1, \, c_1 {>}$ and $S_2 \ {<} f_2, \, c_2 {>}$ are two competing answers to a question, and $S \ < F_{cho} >$ is the chosen one. When S_1 and S_2 are the same statement, the one with a higher *confidence* value is chosen, otherwise the one with a higher *expectation* value is chosen.

$J_2 \setminus J_1$	$S_1 \ {<} f_1, \, c_1 {>}$
$S_2 \ {<} f_2, \, c_2 {>}$	$S \ < F_{cho} >$

Table 3.7: The Choice Rule

3.3.3 Truth-value functions in general

A typical inference rule in NAL has the following format:

$$\{premise_1 <f_1, c_1>, \ premise_2 <f_2, c_2>\} \vdash conclusion <f, c>$$

and a truth-value function calculates $< f, c >$ from $< f_1, c_1 >$ and $< f_2, c_2 >$. Alternatively, it can be put into a table (as we have seen previous) where each row and column corresponds to a premise.

The previously defined revision rule is the only inference rule in NAL whose premises and conclusion contain the same statement. Consequently, the evidence of the premises can be directly treated as evidence of the conclusion, and the conclusion, based on accumulated evidence, has a higher confidence value than the premises.

In the other inference rules, the premises and the conclusion are judgments about different statements, so each of them has its own evidence space, and the evidence of a premise cannot be directly used as evidence of the conclusion. Even if a premise and a conclusion have overlapping evidence spaces, evidence in the premise will be counted less in the conclusion. This is the case because according to the semantics of NAL, "amount of evidence" actually measures evidence that *directly* supports the statement. When indirect evidence is recognized, it is turned into direct evidence with a reduced amount (and the detail differs from rule to rule). Consequently, in every inference rules in NAL, except revision, the conclusion always has a lower confidence value than the premises.

I have introduced several uncertainty measurements, and most of them take values from the [0, 1] interval. Even the amount of evidence, which is not defined with this range in general, corresponds to this interval when it is limited to a piece of evidence within a unit amount. Since they cannot be easily interpreted as "probability" as defined in probability theory and statistics, we cannot directly apply an existing theory to guide their calculation in the truth-value functions attached to various inference rules.[9]

The approach used in NARS is to see the values in [0, 1] as extended Boolean values, 0 and 1, and to handle their calculation by extending the Boolean operators, namely "*not*," "*and*," and "*or*."

[9]Arguments for this conclusion will be provided in Chapter 8.

The extended "*and*" and "*or*" are often called *Triangular norm* (T-norm) and *Triangular conorm* (T-conorm), respectively. They are functions defined on real numbers in [0, 1], being commutative and associative, and monotonic in each variable. T-norm has boundary conditions satisfying the truth tables of the Boolean operator "*and*," and T-conorm those of "*or*." [Bonissone and Decker, 1986, Dubois and Prade, 1982, Schweizer and Sklar, 1983].

In this book, these two functions are directly written as $and(x_1, x_2)$ and $or(x_1, x_2)$. Because each is commutative and associative, they can be extended to take an arbitrary number of arguments:

$$and(x_1, \ldots, x_n) = and(and(x_1, \ldots, x_{n-1}), x_n),$$

$$or(x_1, \ldots, x_n) = or(or(x_1, \ldots, x_{n-1}), x_n).$$

The usage of T-norm and T-conorm in NARS is different from that in other approaches [Bonissone and Decker, 1986, Dubois and Prade, 1982], where they are only used to determine the degree of certainty of the *conjunction* and *disjunction* of two propositions, respectively. In NARS, the T-norm function $y = and(x_1, \ldots, x_n)$ is used when a quantity y is *conjunctively determined* by two or more other quantities $x_1, \ldots, x_n$ — that is, $y = 1$ if and only if $x_1 = \cdots = x_n = 1$, and $y = 0$ if and only if $x_1 = 0$ or ... or $x_n = 0$; similarly, the T-conorm function $y = or(x_1, \ldots, x_n)$ is used when a quantity y is *disjunctively determined* by two or more other quantities $x_1, \ldots, x_n$ — that is, $y = 1$ if and only if $x_1 = 1$ or ... or $x_n = 1$, and $y = 0$ if and only if $x_0 = \cdots = x_n = 0$. These functions are not directly about the *conjunction* or *disjunction* of Narsese statements.[10]

Intuitively, a variable y is conjunctively determined by variables x_1, ..., x_n when all the x's are its *necessary factors*, or numerically, if y is never bigger than any of them. Similarly, y is disjunctively determined by $x_1, \ldots, x_n$ when all the x's are its *sufficient factors*, or numerically, it is never smaller than any of them. In this way, T-norm and T-conorm are applied in situations where a quantity is determined by several factors, where we wish the boundary condition to be satisfied, and where no one factor is more important than any of the others.

[10] In Chapter 5, we will see that they are still used for the conjunctions and disjunctions of statements. They are just not *merely* used in that situation.

There are an infinite number of ways of numerically satisfying the prescribed conditions on T-norm and T-conorm. For our purpose, it is desired for them to be continuous and strictly increasing, so that any upward (downward) change in any argument will cause an upward (downward) change in the function value. In [Schweizer and Sklar, 1983] it is proved that all functions satisfying the above conditions are isomorphic to (i.e., can be represented as a monotonic transform of) the "probabilistic" operators:

$$and(x, y) = xy; \;\; or(x, y) = x + y - xy.$$

It is also shown in [Bonissone and Decker, 1986] that only a small finite subset of the infinite set of possible T-norms and T-conorms will produce significantly different results, if we limit our concern to the "finest level of distinction among different quantifications of uncertainty." Among those representative operators in the small subset, the above pair is the only continuous and strict T-norm and T-conorm. These results show that the above T-norm and T-conorm have not been chosen arbitrarily for NARS; although in principle there are other pairs satisfying our requirements, they are usually more complex, and are not significantly different from the above pair.

The above choice is also justifiable in another way. We call quantities *mutually independent* of each other, when given the values of any of them, the remaining ones cannot be determined, or even bounded approximately. This type of mutual independence among arguments is assumed by the probabilistic operators, but not by other representative operators, such as the "min/max" pair used in fuzzy logic [Bonissone and Decker, 1986].

Obviously, to use the probabilistic operators when the mutual independence does not hold (e.g., $x = y$ or $x = not(y)$) leads to counterintuitive results. In the following, the T-norm and T-conorm are only used when the "mutual independence" condition is satisfied. As far as the two premises are not based on overlapping evidence, f_1, c_1, f_2, and c_2 satisfy this requirement, because given the values of any three of them, the value of the last one cannot be determined, or even bounded.

It should be mentioned that though the T-norm and T-conorm used in NARS share intuition and mathematical forms with probabilistic formula, they should not been understood as $and(x, y) = P(x \text{ and } y)$ and

$or(x, y) = P(x \text{ or } y)$, simply because x and y are usually not random variables with probability distribution function P.

As usual, the "*not*" operator on the extended Boolean variable is defined as

$$not(x) = 1 - x$$

In NAL, the truth-value function for most of the inference rules (with the previously defined revision and choice as exceptions) are built by the following steps:

1. To treat all the uncertainty values involved as Boolean variables whose value are either 0 or 1. According to the definition of these uncertainty measurements and the semantics of Narsese, the uncertainty values of the conclusion is determined for each combination of those of the premises.

2. To represent the uncertainty values of the conclusion as Boolean expressions of the the uncertainty values of the premises that satisfy the above boundary conditions. Usually there are infinitely many functions that satisfy the restriction, and the ones accepted are those that are simple and have natural interpretations.

3. To replace the *and*, *or*, and *not* operator in the Boolean function by the T-norm ($and(x, y) = x * y$), T-conorm ($or(x, y) = 1 - (1 - x)(1 - y)$), and Negation ($not(x) = 1 - x$) functions, respectively, so as to get a general function on [0, 1].

4. To rewrite the uncertainty functions as truth-value functions (if they are not already in that form), according to the relationship between truth value and the other uncertainty measurements.

These are the conceptual steps of the design procedure. When truth-value functions are introduced in the following descriptions, the first two steps are often merged, and the last step is often taken implicitly. Since the result of the above Step 2 may be not unique, the above approach of building truth-value functions is not a mathematical proof of the function obtained, and with the progress of the research, the functions have been modified in different versions of NARS in the past, and it may still happen in the future, if negative evidence for the design of these functions is found. As everything else in this theory, these functions are just “the best we can get according to available evidence.”

3.3.4 Syllogistic rules

In term logics, when two judgments share exactly one common term, they can be used as premises in an inference rule that derives an inheritance relation between the other two (unshared) terms. Altogether, there are four possible combinations of premises and conclusions, corresponding to the four figures of Aristotle’s Syllogisms [Aristotle, 1989], three of which are also discussed by Peirce [Peirce, 1931]. They are listed in Table 3.8.

$J_2 \setminus J_1$	$M \rightarrow P$ $<f_1, c_1>$	$P \rightarrow M$ $<f_1, c_1>$
$S \rightarrow M$ $<f_2, c_2>$	$S \rightarrow P$ $<F_{ded}>$	$S \rightarrow P$ $<F_{abd}>$
$M \rightarrow S$ $<f_2, c_2>$	$S \rightarrow P$ $<F_{ind}>$	$S \rightarrow P$ $<F_{exe}>$

Table 3.8: The Syllogistic Rules of NAL-1

The four rules in the table are explained in the following:

1. $\{M \rightarrow P <f_1, c_1>,\ S \rightarrow M <f_2, c_2>\} \vdash S \rightarrow P <f, c>$
   This rule corresponds to Aristotle’s *first figure* and Peirce’s *deduction*.

2. $\{P \rightarrow M <f_1, c_1>,\ S \rightarrow M <f_2, c_2>\} \vdash S \rightarrow P <f, c>$
   This rule corresponds to Aristotle’s *second figure* and Peirce’s *abduction* (and he also called it *hypothesis*).

3. $\{M \rightarrow P <f_1, c_1>,\ M \rightarrow S <f_2, c_2>\} \vdash S \rightarrow P <f, c>$
   This rule corresponds to Aristotle’s *third figure* and Peirce’s *induction*.

4. $\{P \rightarrow M <f_1, c_1>,\ M \rightarrow S <f_2, c_2>\} \vdash S \rightarrow P <f, c>$
   This rule corresponds to the *fourth figure* of Aristotle’s Syllogistic [Bocheński, 1970].

In NAL, the first three rules are named using Peirce's words. The fourth rule is called *exemplification*. The truth-value functions in the table are named by three letters after the corresponding rule. They are built according to the general procedure introduced previously.

Obviously, each pair of premises also derives a judgment "$P \to S$," whose truth value can be determined by one of the four functions.

The *deduction* rule in NAL-1 extends the "rule of transitivity" in NAL-0. For the frequency of the conclusion, f, it is 1 if and only if both premises have frequency 1. As for the confidence of the conclusion, c, it reaches 1 only when both premises have truth values $< 1, 1 >$. Therefore, the Boolean function we get for deduction is

$$f = and(f_1, f_2), \quad c = and(f_1, c_1, f_2, c_2)$$

which leads to truth-value function

$$F_{ded}: \; f = f_1 f_2, \quad c = f_1 c_1 f_2 c_2$$

The deduction rule is symmetric to the premises, that is, their order does not matter.

In NARS, *abduction* is the inference that, from a shared element M of the *intensions* of S and P, determines the truth value of "$S \to P$," and *induction* is the inference that, from a shared element M of the *extensions* of S and P, determines the truth value of "$S \to P$." Therefore, derived from the duality of extension and intension, we have a duality of abduction and induction in NAL.

In both cases, the premises provide a piece of positive evidence with a unit amount if and only if both of them have truth-value $< 1, 1 >$, which can be represented as Boolean function

$$w^+ = and(f_1, c_1, f_2, c_2)$$

For the total amount of evidence, in abduction we get

$$w = and(f_1, c_1, c_2)$$

and in induction we get

$$w = and(c_1, f_2, c_2)$$

Please note that in the above representation we are mixing two forms of uncertainty measurement: in the premises, the truth values are used, while in the conclusion, the amounts of (positive/total) evidence are used. Also, in these two rules, the two premises play different roles, and their order matters.

After rewriting the result as truth-value functions, for abduction, it is

$$F_{abd}: \ f = f_2, \ c = f_1 c_1 c_2/(f_1 c_1 c_2 + k)$$

and for induction, it is

$$F_{ind}: \ f = f_1, \ c = c_1 f_2 c_2/(c_1 f_2 c_2 + k)$$

In the process of designing truth-value function for induction (and abduction), a crucial point is to see that when the premises are $\{M \to P, M \to S\}$, it is the term M (as a whole) that is taken as evidence, and the amount of evidence it can provide is less than 1. A mistake easy to make here is to think of M as a set of evidences for "$S \to P$," and think of the number of instances in M as the amount of evidence of the conclusion. That interpretation is inconsistent with the semantics of Narsese.

In term logics, "conversion" is an inference from a single premise to a conclusion by interchanging the subject and predicate terms of the premise [Bocheński, 1970]. Now we can see conversion, defined in Table 3.9, as a special case of abduction by taking "$P \to S < f_0, c_0 >$" and "$S \to S < 1, 1 >$" (a tautology) as premises, and "$S \to P < f, c >$" as conclusion.

Using F_{abd}, we obtain the truth-value functions for the conversion rule

$$F_{cnv}: \ f = 1, \ c = f_0 c_0/(f_0 c_0 + k)$$

$$\boxed{\{S \to P \ < f_0, \, c_0 >\} \vdash P \to S \ < F_{cnv} >}$$

Table 3.9: The Conversion Rules of NAL-1

We could also derive this same result by seeing conversion as a special case of induction with "$P \rightarrow P < 1, 1 >$" and "$P \rightarrow S < f_0, c_0 >$" as premises.

Similarly we can get the truth-value functions for *exemplification.* This rule takes the same premises as the deduction rule, but in its conclusion the most general term in the premises, P, becomes the subject, while the most specific term in the premises, S, becomes the predicate. As in the case of the conversion rule, no negative evidence for the conclusion can be collected in this way, and

$$w = w^+ = and(f_1, c_1, f_2, c_2)$$

Therefore the truth-value function is

$$F_{exe}: \ f = 1, \ c = f_1 c_1 f_2 c_2 / (f_1 c_1 f_2 c_2 + k)$$

As mentioned above, from "$M \rightarrow P < f_1, c_1 >$" and "$M \rightarrow S < f_2, c_2 >$," NARS can directly get "$S \rightarrow P < f_1, \ c_1 f_2 c_2 / (c_1 f_2 c_2 + k) >$" by induction. Now there is also an indirect way to derive "$S \rightarrow P$" from the same premises: via conversion, the second premise yields "$S \rightarrow M < 1, c_2 f_2 / (c_2 f_2 + k) >$"; then, deductively combining this judgment with the first premise, NARS arrives at the conclusion "$S \rightarrow P < f_1, \ f_1 c_1 f_2 c_2 / (c_2 f_2 + k) >$." Compared with the direct result, this indirect conclusion has the same frequency value, but a lower confidence value. Similarly, abduction can be replaced by conversion-then-deduction, and exemplification by conversion-then-deduction or deduction-then-conversion, but all of them give lower confidence values, compared to the ones produced by the above rules.

These results show that each application of a syllogistic rule in NARS will cause some information loss (while preserving other information, of course), and therefore *direct* conclusions will always be more confident. On the other hand, the fact that exactly the same frequency value is arrived at by following different inference pathways shows that the truth-value functions defined above have not been coined individually in *ad hoc* ways, but are closely related to each other, since all of them are based on the same semantic interpretation of the truth value.

In general, from the same set of premises, different sequences of inference steps may assign different truth values to the same statement. According to the experience-grounded semantics, the truth value assigned to a statement reflects the evidence collected in a certain way, as specified by the rule used in this step of inference. Therefore, it is normal if different paths lead to different judgments. As far as each step is justified according to the semantics, all the judgments are valid, and the system usually just chooses the most confident one.

By comparing the inference rules of NAL-1, we can get the following conclusions:

- Both frequency and confidence contribute to inference, but in different ways.

- Revision is the only rule where the confidence of the conclusion may be higher than those of the premises.

- The confidence of a syllogistic conclusion is never higher than the confidence of either premise, that is, confidence "declines" in syllogistic inference.

- In general, confidence declines much slower in deduction than in induction and abduction.[11] In deduction, if both premises have a confidence value of 1, the conclusion may also have a confidence value of 1. In induction and abduction, however, the confidence of the conclusion has an upper bound $1/(1 + k)$, far less than 1. So, by saying that "Induction and abduction are more uncertain when compared with deduction," what is referred to is not the "first-order uncertainty," f (inductive and abductive conclusions can have a frequency of 1 when all available evidence is positive), but the "higher-order uncertainty," c.

Here we can see another function of the personality parameter k: to indicate the relative confidence of abductive/inductive conclusions.

[11] This conclusion does not apply to situations where the confidence values of the premises are all very low. However, the conclusions produced in those situation usually have little impact on the system.

Intuitively speaking, all intelligent systems (human and computer) need to maintain a balance between the strictness of deduction and the tentativeness of induction and abduction. Comparatively speaking, a system with a small k relies more on abduction and induction, while a system with a large k relies more on deduction. There is no single "optimal value" for such a parameter, at least for our current discussions.

3.3.5 Backward inference

The inference rules introduced before (except the choice rule) are for *forward inference*, since each of them takes a pair of judgments as premises (except the conversion rule, which takes a single premise), and derive a new judgment as conclusion. *Backward inference*, on the other hand, happens when a judgment and a question are taken as premises. We already discussed a special case of backward inference, that is, the choice rule. This rule is used to decide whether a question can be directly answered by a judgment, as well as to select an answer among candidates.

Formally, the backward inference rules for questions are determined by the following principle: A question Q and a judgment J will give rise to a new question Q' if and only if an answer for Q can be derived from an answer for Q' and J, by applying a forward inference rule.

For example, the system is asked to decide the truth value for "*goose* $\rightarrow$ *swimmer*," that is, whether a goose swims. The system does not have a direct answer for it, but it has a belief "*goose* $\rightarrow$ *bird* $< 1, 0.9 >$." From the question and the belief, a backward inference rule produces a derived question "*bird* $\rightarrow$ *swimmer*," because from an answer to this question and the belief, the system can derive an answer to the original question by deduction.

Defined in this way, it is easy to get backward inference rules from forward inference rules. For example, for a given forward-inference rule table, first we take the conclusions in the table as questions (Q), one premise (J_2) as a judgment (J), and the other premise (J_1, without truth value) as the derived question. After renaming the terms and rearranging the order, we get a backward-inference rule table, in which some terms in the questions can be a "?," indicating a query for terms satisfying given condition.

$J \setminus Q$	$M \rightarrow P$	$P \rightarrow M$
$S \rightarrow M$	$S \rightarrow P$	$S \rightarrow P$
$M \rightarrow S$	$S \rightarrow P$	$S \rightarrow P$

Table 3.10: The Backward Syllogistic Rules of NAL-1

For the forward syllogistic rules in Table 3.8, the corresponding backward-inference rules are in Table 3.10.

This table turns out to be identical to Table 3.8, if the truth-value functions and the question/judgment difference are ignored. This elegant symmetry reveals an implicit property of the syllogistic rules of NARS — that is, for any three judgments J_1, J_2, and J_3, if J_3 can be derived from J_1 and J_2 by a syllogistic rule, then from J_3 and J_1 J'_2 can be derived, which has the same statement as J_2 (their truth values may be different). Intuitively, the three inheritance relations constitute a triangle from any two sides of which the third side can be derived. Such a property does not give rise to infinite loops in the system, because if J_3 is really derived from J_1 and J_2, it must share serial numbers with each of the two, which prevents the system from taking J_3 and J_1 (or J_2) as premises in further inferences.

In NAL-1, if a question cannot be directly answered by the choice rule, backward inference is used to recursively "reduce" the question into derived questions, until all of them have direct answers. Then these answers, together with the judgments contributed in the previous backward inference, will derive an answer to the original question by forward inference.

3.3.6 NAL-1 summary

Now we have completed the description of a Non-Axiomatic Logic, NAL-1, with its formal language, semantics, and inference rules. Let me use an example to show what this logic can do.

To make the description simple, for the initial knowledge we give frequency value 1 to positive judgments, frequency value 0 to negative judgments, and confidence value 0.9 to every judgment. At the beginning, the following judgments are given to the system as experience:

$$\begin{array}{l}
(1)\ swan \rightarrow bird <1, 0.9> \\
(2)\ swan \rightarrow swimmer <1, 0.9> \\
(3)\ seagull \rightarrow bird <1, 0.9> \\
(4)\ seagull \rightarrow swimmer <1, 0.9> \\
(5)\ robin \rightarrow bird <1, 0.9> \\
(6)\ robin \rightarrow swimmer <0, 0.9> \\
(7)\ goose \rightarrow bird <1, 0.9> \\
(8)\ dolphin \rightarrow swimmer <1, 0.9>
\end{array}$$

With (1) and (2) as premises, the induction rule derives

$$(9)\ bird \rightarrow swimmer <1, 0.45>$$

Similarly, from (3) and (4), by induction the system gets

$$(10)\ bird \rightarrow swimmer <1, 0.45>$$

Since (9) and (10) are derived from distinct bodies of evidence, they can be used as premises of the revision rule to get

$$(11)\ bird \rightarrow swimmer <1, 0.62>$$

Given the symmetry of the premises, following the same path the system can get another conclusion

$$(12)\ swimmer \rightarrow bird <1, 0.62>$$

However, this symmetry does not apply to negative judgments. From (5) and (6), by induction the system gets

$$(13)\ bird \rightarrow swimmer <0, 0.45>$$

but its symmetric conclusion gets a confidence value 0. Again, applying the revision rule to (11) and (13), the result is

$$(14)\ bird \rightarrow swimmer <0.67, 0.71>$$

Therefore, the inductive conclusions are just like statistical conclusions, except that they are revised incrementally, under the influence of their confidence values.

What makes NAL different from a purely statistical inference system is that in it different types of inference are unified, and therefore the conclusions are not statistical in the traditional sense anymore. From (7) and (14), by deduction the system get

$$(14)\ goose \rightarrow swimmer < 0.67, 0.43 >$$

but the conclusion does not mean that "Sixty-seven percent of geese can swim." Similarly, from (8) and (14), by abduction the system gets

$$(12)\ dolphin \rightarrow bird < 1, 0.3 >$$

but the conclusion is not based on any observed dolphin which is also a bird. Instead, the evidential support in the conclusion comes from the experienced common property of the two terms.