(1) Constructing isolation logic, and unifying dialectical logic and formal logic

(1.1) Defining four values

Isolation logic has four values: true, false, T | F, and indeterminate. I believe the operations can be conducted as follows: true and false, as opposites, are unified into 'indeterminate'. By negating 'indeterminate', the determinate 'true or false' (T | F) can be obtained. Expressed symbolically:

Where, T = true, F = false, T | F = true or false, Ind = Indeterminate.

1) (T,F) represents the opposition of true and false.

2) unify(T,F) = Ind

Since true and false are in opposition, through the negation of this opposition, true and false are unified as indeterminate. It indicates that the two opposing sides, under certain conditions, exist in an indeterminate state. This operation is logically similar to a decision-making process or a quantum superposition state.

3) not(Ind) = T | F

By negating 'indeterminate', a determinate 'true or false' is obtained. This operation is equivalent to 'selecting' a determinate value from an indeterminate state, returning from an indeterminate state to classical two-valued logic. In logic, this process is similar to the transformation from a fuzzy state to a clear state, or similar to the quantum collapse caused by measurement. Here, 'not' negates 'indeterminate'.

4) not unify(Ind) = (T,F)

'not unify' is 'oppose'. This operation transforms the unity of indeterminacy into the opposition of true and false. Here, 'not' negates 'unify'.

(1.2) Constructing a Three-Valued Lattice Using Lattice Theory

It is stipulated that in this lattice operation, Ind will automatically execute the operation not(Ind) = T | F.

1) Let our set of logical values be L = {T, F, T | F}, which includes three values:

T represents true;

F represents false;

T | F and F | T are not distinguished in the lattice operation.

Define a partial order relation ≤ such that it satisfies: F ≤ T | F ≤ T.

2) Lattice operations:

Meet (∧): Defined as the AND operation in logic, but extended here to a three-valued operation:

T ∧ T = T

T ∧ F = F

T ∧ Ind = T ∧ not(Ind) = T ∧ (T | F) = T | F: Since Ind is indeterminate, it will become either T or F. The meet of T with these two cases is T and F, respectively. Therefore, T ∧ Ind = T | F.

F ∧ F = F

F ∧ Ind = F ∧ (T | F) = F: Since Ind is indeterminate, it will become either T or F. The meet of F with these two cases is F and F, respectively, which is determinate. Therefore, F ∧ Ind = F.

Ind ∧ Ind = (T | F) ∧ (T | F) = T | F

This operation is an extension of the 'AND' operation in formal logic, but considering the complexity of isolation logic, we have defined the relationship between indeterminate and other values to ensure consistency in the operation.

Join (∨): Defined as the OR operation in logic:

T ∨ T = T

T ∨ F = T

T ∨ Ind = T ∨ (T | F) = T: Since Ind is indeterminate, it will become either T or F. The join of T with these two cases is T and T, respectively, which is determinate. Therefore, T ∨ Ind = T.

F ∨ F = F

F ∨ Ind = F ∨ (T | F) = T | F: Since Ind is indeterminate, it will become either T or F. The join of F with these two cases is T and F, respectively. Therefore, F ∨ Ind = T | F.

Ind ∨ Ind = (T | F) ∨ (T | F) = T | F

This operation is an extension of the 'OR' operation in formal logic, and it handles the interaction among true, false, and indeterminate.

3) Negation Operation:

'not' in formal logic:

not(T) = F

not(F) = T

not(T | F) = T | F

'not' in dialectical logic:

not(Ind) = T | F

not unify(Ind) = (T, F)

4) Unify Operation:

unify(T, F) = Ind

unify(F, T) = Ind

unify(X, X) = X for X in {T, F, Ind}: unify(X, X) should be equivalent to unify(X), which can be understood as 'the unity of a single entity'. It indicates that an entity, in the absence of an external opposite, has a self-consistent unity. In other words, X itself is unified; to unify X itself results still in X. This is, in fact, an operation on identity, because X itself possesses identity. Therefore, this unify(X) is equivalent to an identity operation, and its meaning should be 'X is X', which is different from a unify operation with two different opposing parameters. For the operation unify(X), the result is X, emphasizing the intrinsic unity and identity of X. This operation does not involve the unification of opposites, but rather the confirmation of the state of X itself. Thus, unify(X, X) can be abbreviated as unify(X).

unify(T, Ind) = F: Intuitively, the result of unifying T and Ind should be an indeterminate state. However, according to isolation dialectical logic, this unify operation requires that T and Ind be in total opposition; that is, negating T must yield Ind, and vice versa. But this cannot be done. In other words, T and Ind do not form an opposition, so the value of unify(T, Ind) is false (F). This seemingly counter-intuitive result precisely embodies the rigor of isolation logic: the 'unify' operation itself has strict conditions for its application, namely that its objects must be true logical opposites.

unify(F, Ind) = F: The same logic as unify(T, Ind) = F.

unify(T, T | F) = F: The same logic as unify(T, Ind) = F.

unify(F, T | F) = F: The same logic as unify(T, Ind) = F.

unify(T | F, F | T) = Ind: T | F cannot have a determinate independence like T does. Therefore, unify(T | F, T | F) cannot equal T | F. unify(T | F, T | F) should be seen as: unify(T | F, F | T) (or unify(F | T, T | F)). Because the negation of T | F is F | T (T | F will become either T or F; if it is T, then its negation is F; if it is F, then its negation is T). Therefore, the two parameters are in opposition: negating one side yields the other. Consequently, unify(T | F, F | T) = Ind.

From the perspective of dialectical logic, not(T | F) does indeed change the result. But this change is meaningless for formal logic, because T | F is indeterminate; we do not know if it has been changed. This is the difference between the two logics. Therefore, in formal logic, not(T | F) = T | F is not a problem, and T | F and F | T can be left undifferentiated. But in dialectical logic, this is not the case. In dialectical logic, they are in a state of relative isolation.

This shows that there is a subtle but crucial difference between these two parameters. Thus, when ignoring the difference between the two parameters, we can abbreviate unify(T | F, T | F) = Ind to unify(T | F) = Ind. Why abbreviate it? Because in operations, a situation with two identical parameters like unify(T | F, T | F) will occur. At that point, it needs to be transformed into a mode where the parameters are in opposition. Abbreviating it to unify(T | F) = Ind is to reduce this trouble. This is, of course, on the condition that it does not cause misunderstanding.

Ind is the state after unification; it is indeterminacy expressed by a determinate value. Whereas T | F is the acquisition of a determinate value; it is the destruction of the unified state, but it is expressed as an uncertainty as to which value is obtained. It is the use of uncertainty to obtain a determinate value. Therefore, this acquisition of determinacy in T | F does not eliminate the underlying indeterminacy and opposition. Consequently, one can use unify(T | F) = Ind to once again return this uncertainty and opposition to the indeterminacy of opposition.

In pure lattice operations, Ind will automatically execute the operation not(Ind) = T | F. This is because all lattice operations are operations of formal logic (even though this lattice has three values), and it is only when performing dialectical logical operations that a conversion to Ind (unify(T | F) = Ind) is required. In this way, the operations of the two logics are separated, avoiding confusion. In the operations of formal logic, the value Ind cannot be operated on directly, because it is a value in dialectical logic. It can only be operated on by being transformed into T | F in formal logic. This is like not being able to directly operate on the quantum state of the quantum world from within the macroscopic world; it must be transformed into a classical value through measurement.

In other words, in the operations of a pure lattice, it is still a three-valued operation, still a three-valued lattice. In this way, the operations between formal logic and dialectical logic do not interfere with each other, yet they are interconnected. This also allows us to see that in formal logic, an intermediate value Ind between true and false does not exist; it is only in dialectical logic that an intermediate value Ind exists between true and false. Although this value can be called an intermediate value, this intermediate value is the value of the unification of true and false. Therefore, this intermediate value is not a simple linear value that appears between true and false. Ind can be seen as a state that can collapse into either T or F. This collapse makes us, in formal logic, think of Ind as a value between T and F.

(1.3) Properties of the Algebraic Structure

After defining these basic operations, we need to ensure that they conform to certain algebraic properties, so that the entire system is self-consistent. To maintain mathematical rigor, we need to verify that our operations satisfy the following properties:

Idempotency: $a \wedge a = a$, $a \vee a = a$

Associativity: $(a \wedge b) \wedge c = a \wedge (b \wedge c)$, $(a \vee b) \vee c = a \vee (b \vee c)$

Commutativity: $a \wedge b = b \wedge a$, $a \vee b = b \vee a$

Absorption: $a \wedge (a \vee b) = a$, $a \vee (a \wedge b) = a$

Distributivity: $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$, $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$

Duality: $\mathrm{not}(a \wedge b) = \mathrm{not}(a) \vee \mathrm{not}(b)$, $\mathrm{not}(a \vee b) = \mathrm{not}(a) \wedge \mathrm{not}(b)$

Through verification, the three-valued operations I have defined satisfy the above properties. Therefore, it is a lattice.

(1.4) De Morgan Lattice

For all possible combinations of the values $p, q \in \{T, F, Ind\}$, it satisfies De Morgan's laws:

$\neg(p \wedge q) = \neg p \vee \neg q$

$\neg(p \vee q) = \neg p \wedge \neg q$

Therefore, we can conclude that the three-valued lattice I have defined is a De Morgan lattice.

This means that this three-valued lattice maintains logical consistency and symmetry when handling negation, meet, and join operations.

A De Morgan lattice is a generalization of a Boolean algebra. Since the three-valued lattice I have defined satisfies duality, it preserves the key structural properties of classical Boolean logic while extending it to handle the additional value 'indeterminate'. This is an ideal feature because it means that this three-valued lattice is not a completely arbitrary system, but is built upon and generalizes the principles of classical logic.

This three-valued lattice satisfies the basic axioms of a lattice (associativity, commutativity, absorption, distributivity, duality, and De Morgan's laws), which shows that it is a mathematically self-consistent and complete structure. This rigor provides a solid foundation for this isolation logic, making it not just a philosophical speculation, but a system that can be formally verified.

In this way, an isolation logic system has been defined that includes a three-valued lattice and dialectical logic. This isolation logic in fact combines and unifies dialectical logic and formal logic, forming a unified logic. This unification is 'perfect' because it not only preserves the advantages of both but also compensates for their respective limitations.

This method can also be seen as extending traditional formal logic to isolation logic, providing new tools for handling uncertainty and ambiguity. This is a natural extension of two-valued logic (true, false). This extension not only does not destroy the law of the excluded middle (that is, that every proposition can only be true or false), but rather, by introducing indeterminacy (Indeterminate), it extends the range of truth values of the logic, enabling the logical system to express more complex states. It not only extends classical two-valued logic but also maintains the core principles of formal logic.

This method provides a logical foundation for understanding the uncertainty in the real world. For example, in scientific research, the uncertainty of measurement, probabilistic events, or unknown variables can be more naturally expressed and handled within this logical framework.

This isolation logic is not just a logical system; it provides a mathematized mode of expression for dialectical logic. Through the unify and not operations in isolation logic, it is possible to mathematically simulate how true and false are mutually transformed through opposition and unity, thereby providing support for the modes of thought in dialectical logic. This makes dialectical logic no longer just a philosophical concept, but something that can be realized within a formalized logical framework.

(2) This isolation logic will change the way people think about problems. Here are a few examples:

(2.1) An Isolation Logic Analysis of the Trolley Problem

The Trolley Problem is a classic ethical thought experiment, proposed by Philippa Foot in 1967 (Foot, "The Problem of Abortion and the Doctrine of the Double Effect," 1967).

Description:

A madman has tied five innocent people to a trolley track. An out-of-control trolley is heading towards them and will run them over in a moment. Fortunately, you can pull a lever that will switch the trolley to another track. The problem, however, is that the madman has also tied one person to the other trolley track. This would sacrifice the other person on that track.

The problem you face:

In this situation, should you pull the lever?

1) Traditional Two-Valued Logic Analysis:

Within the framework of traditional two-valued logic, we must make an either/or choice between pulling the lever and not pulling it.

If you choose to pull the lever (T - True):

Argument: Sacrificing one person to save the lives of five aligns with the principle of utilitarianism: maximizing happiness or minimizing suffering.

Conclusion of two-valued logic: The statement 'pull the lever' is true (T).

If you choose not to pull the lever (F - False):

Argument: Pulling the lever is an active intervention that directly leads to the death of one person, whereas standing by is merely a passive acceptance of fate's arrangement, carrying less moral responsibility.

Conclusion of two-valued logic: The statement 'pull the lever' is false (F), and 'do not pull the lever' is true (T).

2) The Dilemma of Traditional Two-Valued Logic:

Traditional two-valued logic forces us into a black-and-white choice between 'pull' or 'not pull', which leads to a black-and-white binary opposition and is unable to express the uncertainty that exists in the ethical dilemma.

3) The Solution of Isolation Logic:

By applying isolation logic, we can transcend the limitations of traditional two-valued logic, appropriately express the trolley problem, and analyze it more profoundly.

Analysis using isolation logic:

unify(T, F) = Ind: This acknowledges the ethical uncertainty of the trolley problem. Both 'pulling the lever is true' (from a utilitarian perspective) and 'pulling the lever is false' (from a moral responsibility perspective)—these two 'opposing viewpoints'—have a certain 'reasonableness' ethically, but both also have limitations. By 'unifying' (unify) these two opposing viewpoints, we obtain a conclusion that is 'ethically indeterminate (Ind)'. This appropriately expresses our true feeling when faced with an ethical dilemma: being in a quandary and finding it difficult to choose.

Although the trolley problem is ethically 'indeterminate', this does not mean that we can refrain from making a decision. Negating 'indeterminate' (not(Ind)) means that we must make a choice between true (T) and false (F) (T | F). This choice can be made based on different ethical principles, value judgments, or specific contexts. For example:

Based on the principle of utilitarianism: Choose to pull the lever (T): to maximize the happiness of the majority, sacrificing one person to save five.

Based on the principle of moral responsibility: Choose not to pull the lever (F): to uphold the moral responsibility of not actively harming others, not pulling the lever is the more moral choice.

Based on other principles or contexts, different 'choices' may also be made.

The Advantages of Isolation Logic:

Accommodates the 'indeterminacy' of ethical dilemmas: Isolation logic can directly express and handle the inherent ethical uncertainty in the trolley problem, acknowledging that ethical judgments are not always black and white, but that gray areas and ambiguity exist.

Shows the 'necessity of choice': Even when faced with ethical uncertainty, isolation logic still emphasizes the 'necessity of choice'. Through the not(Ind) operation, it highlights the 'responsibility and commitment' to make a decision in an ethical dilemma.

Compared to the simplification and absolutism of traditional two-valued logic, the analysis of isolation logic is more appropriate and more humane, and closer to our real ethical experiences and emotions. It acknowledges the uncertainty of ethical judgment and also respects the 'reasonableness' of different ethical principles and value judgments.

(2.2) The Judge's Verdict in a Legal Case

A judge is presiding over a complex case in which the defendant, Mr. S, is accused of insider trading. However, the evidence is not clear-cut; there are factors that point to guilt as well as factors that point to exoneration.

Evidence for conviction (the possibility of True - T):

Mr. S conducted trades before a major company announcement that had a significant impact on the stock price.

There was an unusual pattern of communication between Mr. S and company insiders.

Evidence for exoneration (the possibility of False - F):

Mr. S claims that his trades were based on independent market analysis, not inside information.

Formal Logic (Applying legal rules and evidence):

Propositions (the strength of formal logic): The judge first isolates the key propositions and evaluates them according to formal legal rules and established legal precedents.

Definitions:

P1: "Mr. S conducted trades before a major company announcement." (Evaluated as True - T based on factual records)

P2: "There was unusual communication between Mr. S and company insiders." (Evaluated as True - T based on communication records)

P3: "Mr. S's trades were based entirely on inside information." (Evaluated as Indeterminate - Ind). This is the key point of uncertainty. Although there is evidence suggesting insider trading (P1 and P2), there is no conclusive evidence to establish P3 as true beyond a reasonable doubt, which is the legal standard.

P4: "Mr. S's trades were entirely innocent, based on independent analysis." (Evaluated as Indeterminate - Ind). Although Mr. S has offered this explanation, the judge cannot definitively rule out the possibility of insider trading based on the existing evidence.

Using Meet (∧) to evaluate the combination of evidence:

"P1 ∧ P2 ∧ P3" (Trading before announcement AND Unusual communication AND Trades based entirely on inside information).

Evaluation using Meet: T ∧ T ∧ Ind = T | F.

Interpretation of Meet (∧) in Isolation Logic: Although P1 and P2 are true, the entire prosecution case, represented by their conjunction with P3 (indeterminate), becomes indeterminate. This reflects that while there is some evidence pointing to guilt, the indeterminacy of P3 weakens the entire case, leaving it in a state of uncertainty. Formal logic here highlights the weakest link in the chain of evidence.

Dialectical Logic:

Identifying Opposing Arguments (the domain of dialectical logic):

Thesis (Prosecution's argument - Guilty): Based on the available evidence, Mr. S is likely guilty of insider trading.

Antithesis (Defense's argument - Not Guilty): Based on the lack of conclusive evidence and alternative explanations, Mr. S is likely not guilty of insider trading.

Applying Unify(T, F) = Ind to Acknowledge Inherent Uncertainty: The judge recognizes that based on the available evidence, neither side presents a clear conclusion. They are mutually opposing views on an essentially indeterminate situation. Therefore, the judge applies the unify operation to acknowledge this dialectical tension and inherent uncertainty:

unify(Thesis: Guilty (T), Antithesis: Not Guilty (F)) = Ind

Interpretation of the Unify operation: The judge concludes that based on the current state of the evidence and the opposing arguments, the final verdict is 'indeterminate'. This is not a decision of evasion or failure, but a logically sound and philosophically nuanced acknowledgment of the epistemological limitations and the inherent ambiguity of the situation. In this specific legal context, based on the available information, the truth remains unascertainable.

Applying Not(Ind) = T | F to Reach a Verdict (Making a choice under uncertainty): However, the judge cannot simply remain in a state of 'indeterminate'. The legal system requires a clear verdict. Therefore, the judge must now apply the not(Ind) operation to make a 'choice' between true (guilty) and false (not guilty), based on the available information and prevailing legal principles:

not(Ind: Verdict is indeterminate) = Guilty (T) | Not Guilty (F)

Interpretation of the not(Ind) operation: After negating 'indeterminacy', the judge must make a judgment. This choice is not arbitrary but is guided by legal principles and the weight of the evidence (even if the evidence is not conclusive):

Possible Choice 1: Not Guilty (F): If the judge prioritizes the principle of 'presumption of innocence' and the higher standard of proof in criminal cases, they might choose not guilty (F). This 'choice' leans towards protecting individual rights in the face of uncertainty.

Possible Choice 2: Guilty (T): In a different legal system, or with a slightly different weighting of the evidence, a judge might choose guilty (T), perhaps emphasizing the seriousness of insider trading and the need to deter such behavior, even without absolute certainty.

This example shows how isolation logic, by combining formal and dialectical elements, provides a powerful and nuanced tool for analyzing complex situations involving uncertainty, opposition, and the need for choice, far surpassing the limitations of traditional binary logic.

(2.3) Paradoxes

1) The Liar Paradox: 'This sentence is false' (P). If this sentence is true, then it is false; if it is false, then it is true (Tarski, "The Concept of Truth," 1933). Traditional two-valued logic cannot handle this type of self-reference because it leads to a logical contradiction.

This sentence can be regarded as Ind. Because it simultaneously expresses two opposing meanings, we can therefore say that the truth or falsity of this sentence cannot be determined, thus avoiding a logical self-contradiction. By regarding it as indeterminate, we can see that this is a vague, undecidable proposition, not a simple opposition.

2) The Barber Paradox: In a town, the barber announces that he only shaves those who do not shave themselves. So, should this barber shave himself? If he shaves himself, it is a contradiction. If he does not shave himself, then he can shave himself. These are two opposing results, so this paradox can be seen as indeterminate.

3) Russell's Paradox: Proposed by Bertrand Russell (Russell, Principles of Mathematics, 1903). Consider a set R defined as "the set of all sets that are not members of themselves." That is: $R = \{x \mid x \notin x\}$ (the set x does not contain itself).

Proposition: The set R contains itself ($R \in R$).

Analysis:

If $R \in R$ = True, then $R \notin R$ (according to the definition), which is a contradiction.

If $R \in R$ = False, then $R \in R$ (satisfies the definition), which is a contradiction.

Similarly, one can set $R \in R$ = Ind, indicating that 'whether a set can contain itself' is indeterminate.

Here, it reflects not only a logical contradiction but also implies the mathematical indeterminacy or indefinability of the concept 'a set containing itself'. From a formal structural point of view, the Liar Paradox and Russell's Paradox are the same. However, the barber can shave himself, but whether 'a set can contain itself' is indeterminate. This is the essential difference between these two paradoxes. In fact, in the subsection "Formal Logic," it has already been argued that 'a non-empty set cannot contain itself'. Isolation logic correctly expresses this indeterminacy.

Unlike classical logic, isolation logic does not collapse or lead to contradiction when faced with paradoxes. It provides a logically consistent way to contain and analyze paradoxes within the system itself.

References

Foot, P. (1967). The Problem of Abortion and the Doctrine of Double Effect. Oxford Review, 5, 5-12.

Tarski, A. (1933). The Concept of Truth in Formalized Languages. Translated by J. H. Woodger, Logic, Semantics, Metamathematics, Oxford University Press, 1956.

Russell, B. (1903). Principles of Mathematics. Cambridge University Press.