Jan Pociej
Philosopher

SOLVING THE ROSS AND PRIOR PARADOXES.
CLASSICAL MODAL CALCULUS

ABSTRACT: Resolving the Ross and Prior paradoxes proved to be a difficult task. Its first two stages, involving the identification of the true natures of the implication and truth values, are described in the articles *Solving the Paradox of Material Implication – 2024* and *Solving Jörgensen's Dilemma – 2024*. This article describes the third stage, which involves the discovery of missing modal operators and the Classical Modal Calculus. Finally, procedures for solving both paradoxes are provided.

KEYWORDS: solution, Ross, Prior, paradox, logic, classical, imperative, deontic, values, philosophy, metaphysics, sentences, affirmative, evaluative, normative, evaluations, norms, command, prohibition, permission, obligation, declaratives, state of affairs, implication, competition, opposition, geometry of oppositions, exclusive, inclusive, functionality, calculus, propositional, modal

1. The Ross paradox and the Prior paradox

Opponents of non-declarative logic believe that only declarative sentences, i.e. declaratives, are sentences in the logical sense. To support their position, they cite the Ross paradox, formulated by the Danish lawyer and legal philosopher Alf Niels Christopher Ross (1899-1979), and the Prior paradox, formulated by the New Zealand logician Arthur Norman Prior (1914-1969).

The strength of Ross's paradox comes from the recognition of the rule of disjunctive amplification as an implication, which in the Classical Propositional Calculus (CPC) has the form $p \Rightarrow p \vee q$ (or $q \Rightarrow p \vee q$), and in deontic logic the form $OA \Rightarrow O(A \vee B)$. The deontic formula says that if a given action is obligatory, then the given action or some other is obligatory. For example, if the variable "A" is defined as "Send a letter!", the implication leads to the conclusion "Send the letter or burn it!", which is unacceptable (Ross 1941).

Prior's paradox concerns the formula of deontic logic O~A=>O(A=>B). If it is read as an implication, one must agree to the reasoning according to which committing a prohibited act obliges the perpetrator to do something more – for example, committing theft also obliges him to commit adultery. Von Wright found Prior's observation a "real difficulty" (Von Wright 1956) and, in order to solve this paradox, he created the first system of dyadic deontic logic, which gave rise to a whole galaxy of similar solutions, but without the expected result.

Describing the paradoxes of Ross and Prior, the Swedish logician Jörg Hansen states that "since these paradoxes have troubled deontic logic for three generations and called the whole enterprise of deontic logic into question, a solution would be extremely welcome" (Hansen 2006).

2. Solution to the Ross paradox

The key to solving Ross's paradox is – firstly – the correct identification of the nature of the logical operation known since antiquity as implication, and – secondly – the recognition of imperatives as sentences in the logical sense.

The identification of the nature of the alleged implication was made in the article S*olving the Paradox of Material Implication – 2024* (Pociej 2024/1). It turned out that the implication is actually a kind of opposition, for which the name "competition" was proposed. Replacing the implication with the competition required an appropriate change of the connective, so instead of the connective "if... then" the conjunction "but" was proposed, and because it turned out that some sentences of the competition sound better with the use of other opposing conjunctions, such as “neverthless”, "alternatively” or “rather…than”, their use has also been proposed.

The fact of logical sense of imperative sentences – and in general of all sentences except self-sentences – was demonstrated in the article *Solving Jörgensen's Dilemma* (Pociej 2024/2).

Based on the above-mentioned philosophical decisions, the formula p=>(p∨q) can be read as the following sentence:

PR1: “Send the letter; alternatively, send it or burn it!”

There is nothing paradoxical about such a sentence and therefore it can be considered that the Ross paradox has been resolved on the basis of the CPC. Logicians, however, looked for a solution to Ross's paradox not within the framework of the CPC, but on the basis of deontic logic, because it referred to sentences in the imperative mood. These sentences do not contain modal verbs, but the imperative mood makes their predicates play the role of commands, which are one of the deontic modalities. The obligation operator in the formula OA=>O(A∨B) could therefore mean both a modal verb (e.g. should) and imperative mood. However, a similar ambiguity does not apply to the permission operator. This raises the question of what deontic operators actually mean and what their mutual relations are.

Once again, the analogy between logical functionality and the results of a football match, used in the article *Solving the Paradox of Material Implication – 2024*, helps to find answers to these questions. There, atomic sentences refer to simple events – scoring or not scoring a goal, and their combinations, interpreted in various ways using logical functions, describe the results of the match, such as winning, not losing, etc. By analogy to that analogy, it should be considered that such imperatives, like "Send a letter!" and "Don't send the letter!" mean particular modalities (facts), i.e. commands and prohibitions, and their combinations constitute general modalities (states of affairs), which are equivalents of logical operations. Particular modalities, as mentioned above, do not have to be expressed in separate verbs, but they create general modalities in the same way as the facts of scoring or not scoring goals create the result of the match. Therefore, it can be assumed that between the imperative "Send a letter!" and the normative "You have to send a letter!" there is a relationship similar to that between an individual command and a statement of obligation.

Having established this, let us proceed to solve Ross's paradox on the basis of deontic logic. The starting point will be the second axiom of deontic logic, formulated by Von Wright, having the form O(A=>B)=>(OA=>OB) and read as an implication: "If the implicaation from A to B is valid, then the validity of A implies the validity of B”. Taken as a competition, this axiom would take the form O(A=>B)<=>(OA=>OB) and would mean: "If the competition between A and B holds, then the obligation of A competes with the obligation of B." Consequently, the formula O(A=>(A∨B)) would be transformed into the formula OA=>O(A∨B), which would be read as:

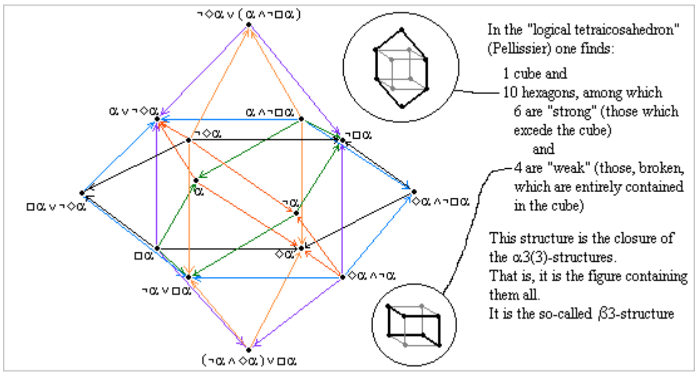
PR2: "You should send the letter; alternatively, you should send it or burn it."

As you can see, the deontic version seems to correspond to the propositional version. However, a question immediately arises regarding the logical values of deontic operators: what exactly are they? How is it that the logical value of the obligation operator of a disjunction seems to be the same as the value of the obligation operator of one of its terms? To answer these questions, we must first determine the set of all deontic operators.

3. Completing the set of deontic operators

Logicians (Lewis, Kripke, Feyes, Von Wright) developed several systems of modal propositional calculus, but their search focused on using a minimum number of operators (most often these are operators of necessity and possibility, which in deontic logic function as operators of command and permission). To create a complete list of deontic operators, opposition geometry turns out to be more useful. Its core is the logical square described by Aristotle, also called the square of opposition. According to this geometry, the oppositions create a structure, called by Alessio Moretti the *β3-structure*, whose spatial counterpart, in the form of a tetraicosahedron, was discovered by the French mathematician Régis Pellissier. The view of this structure is as follows (Moretti 2009, 209):

Figure 1



Vertices denote opposing elements, and edges and diagonals denote relations of opposition (contradiction, contrariety, subcontrariety and subalternation). The opposing elements can be both logical operations (conjunction, disjunction, etc.) and deontic operators or the results of a football match. Oppositions are logical operations: the contravalence corresponds to contradiction (*contradictio*), the non-conjunction to contrariety (*contrarietas*), the disjunction to subcontrariety (*subcontrarietas*) and the competition to subalternation (*subalternatio*).

The geometry of oppositions in connection with the isness table method allows us to establish an analogy between logical functions and deontic operators. Four of these operators, meaning command, prohibition and permission to do something or permission not to do something – marked in the traditional logical square by the letters A, E, I and O – have been known for centuries. However, Pellisier's structure contains six other operators that have yet to be identified. Their identification is aided by an analogy to the results of a football match, as seen in the table below.

Table 1 (Operator Table)

States of affairs related to the match	Symbols of logical	Logical values	Modal operators (here: deontic)

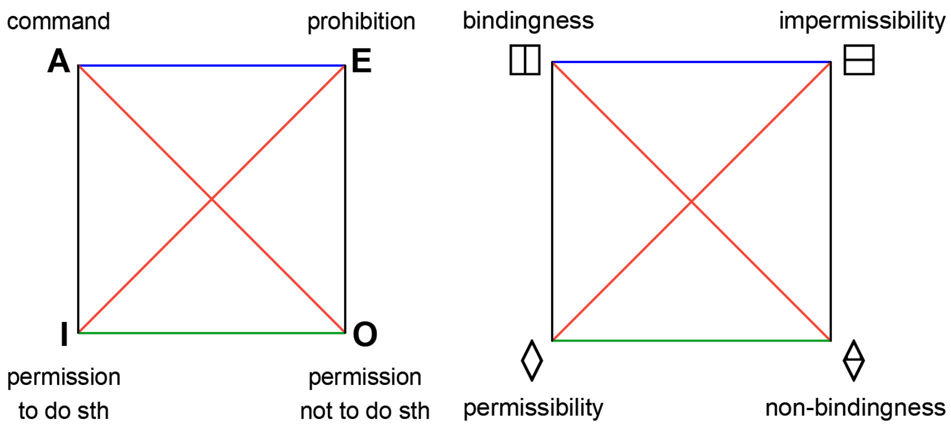
	operations		
A score a goal	A	1100	□A command A (one should A)
A don’t score a goal	~A	0011	⧄A prohibition A (one shouldn’t A)
B score a goal	B	1010	□B command B (one should B)
B don’t score a goal	~B	0101	⧄B prohibition B, (one shouldn’t B)
A win, B lose	A≠>B	0100	◫A, ⊟B bindingness of A, impermissibility of B; (it’s binding to A/not B; it’s impermissible not to A/ to B; one should A, but not B/ not B, but A)
A lose, B win	A<≠B	0010	⊟A, ◫B impermissibility of A, bindingness of B; (it’s binding to B/not A; it’s impermissible not to B/ to A; one should B, but not A/ not A, but B)
A don’t lose, B don’t win	A<=B	1101	◊A, ⟠B permissibility of A, non-bindingness of B; (it’s permissible to A/ not B; it’s omissible to B; one should B, but/neverthless/alternatively should A; one should rather A than B)
A don’t win B don’t lose	A=>B	1011	⟠A, ◊B non-bindingness of A, permissibility of B; (it’s not binding to A; it’s permissible to B/ not A; one should A, but/neverthless/alternatively should B; one should rather B than A)
a draw	A<=>B	1001	○ equality (one should equally A if B)
not a draw	A<≠>B	0110	⊖ non-equality (one should non-equally A if not B)
a goal draw	A∧B	1000	⦶ co-bindingness (one should A and B together)
a goalless draw	A↓B	0001	◎ co-impermissibility (one should both not-A and not-B)

not a goal draw	A↑B	0111	⊕ non-co-bindingness (one shouldn’t A and B together)
not a goalless draw	A∨B	1110	⊜ non-co-impermissibility (one shouldn’t both not A and not B)

The selection of names for some operators may be controversial – as is usually the case with new proposals. Deontic literature talks, on one hand, about such deontic operators as “command”, “prohibition” and “permission”, and on the other, about “obligation” or “bindingness”, “impermissibility”, “permissibility” and “omissibility”. Moreover, the term "order" is used interchangeably with the terms “command” and "obligation", and the term “prohibition” replaces here and there the term “prohibition”. With such abundant terminology at our disposal, all that remains is to choose appropriate names for the operators defined by Boolean functions. The terms "command" and "prohibition" seem to best express simple modalities such as "Do A!" and "Don't do A!" The term “bindingness” (rather than "obligation") should be reserved for the operator corresponding to Boolean inhibition (former non-implication). Since “bindingness” and “permissibility” form the A-I side of the logical square, the E-vertex symbolizing the contrariety of “bindingness” and the negation of “permissibility” should be called "impermissibility". Consequently, the operator corresponding to the vertex O should be given the name "non-bindingness".

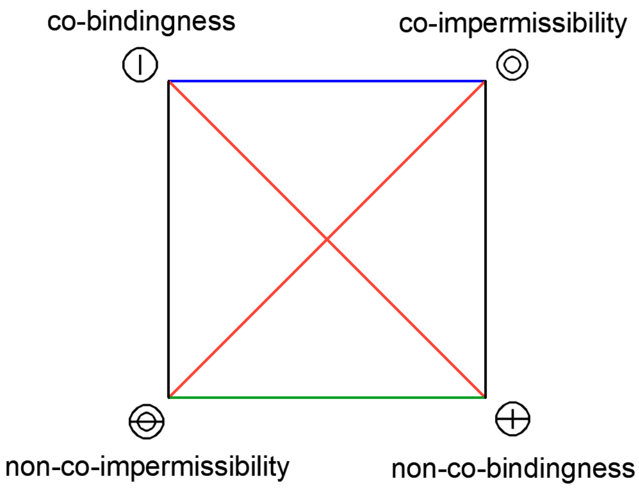
Four operators – bindingness, permissibility, non-bindingness and impermissibility – form a logical square in which command, prohibition and two varieties of permission were traditionally placed. This square remains valid. Permissibility and non-bindingness remain two varieties of permission, while command and prohibition are replaced by bindingness and impermissibility, as can be seen in the illustration below.

Figure 2



An analogous logical square is formed by the operators of co-bindingness co-impermissibility, non-co-impermissibility and non-co-bindingness. Its correctness seems obvious.

Figure 3



The proposed names of the remaining new operators, such as "co-bindingness" or "non-co-impermissibility", are, admittedly, long-winded, but they seem precise enough to conduct further considerations.

4. The Classical Modal Calculus

Of course, the question arises why for centuries there has been no distinction between "command" and "bindingness" or "prohibition" and "impemissibility". The reason seems to be the close affinity of the meanings of these terms – every bindingness is closely related to a command and every impermissibility is closely related to a prohibition. Seemingly, it is only possible to notice the differences between them thanks to the application of modality calculus.

Such a calculus exists and has so far been treated as the Classical Propositional Calculus extended to include the operators of command, prohibition and permission. However, by completing the set of deontic operators and assigning command and prohibition operators roles analogous to sentence variables, the modality calculus becomes the full-fledged Classical Modality Calculus (CMC). Thus, the theorem about the non-existence of truth-functionality – or more precisely, isness-functionality – of modal operators is falsified. This functionality exists and allows semantics to be written in the isness-tables format.

For example, statements such as:

1) “Permissibility is the disjunction of bindingness and equality, just as not losing is the disjunction of winning and drawing.”

and

2) “Non-bindingness is the disjunction of impermissibility and equality, just as not winning is the disjunction of losing and drawing.”

can be written as in the table below.

Table 2

◫A ⊟B	⊟A ◫B	(◫A<=>◫B) <=> ○(A<=>B)	(◫Av○(A<=>B))<=>◊A	(⊟Av○(A<=>B))<=>⟠A
0	0	1	1	1
1	0	0	1	0
0	1	0	0	1
0	0	1	1	1

Similarly, statements such as:

1) "The equality of permissions is the conjunction of the permission to A and the permission to B, just as a draw is the conjunction of not losing and not winning."

and

2) "The conjunction of commands is a co-bindingness, just as an equal number of goals scored means a goal-draw, and the conjunction of prohibitions is a co-impermissibility, just as neither team scoring a goal means a goalless draw."

can be written as in the table below.

Table 3

◊A ⟠B	⟠A ◊B	(◊A∧◊B) <=> ○(A<=>B)	□A	□B	(□A∧□B) <=> ⦶(A∧B)	⧄A	⧄B	(⧄A∧⧄B) <=> ◎(~A∧~B)
1	1	1	1	1	1	0	0	0
1	0	0	1	0	0	0	1	0
0	1	0	0	1	0	1	0	0
1	1	1	0	0	0	1	1	1

The transformations of the formulas in both tables were made based on several rules that should be mentioned here.

First rule: the logical values of variables are identical to the logical values of the command and prohibition operators associated with these variables, which can be written as the first axiom of the Classical Modality Calculus in the form: □X|<=>|X| and |⧄X|<=>|~X|, where "X" stands for a sentence variable. This axiom is the basis for the unity of CPC and CMC.

Second rule: operators of logical operations, hereinafter referred to as operational operators, are equivalent to the results of these operations on variable operators, which can be written as the second axiom in the form F(A–B)<=>(FA–FB), where “F” stands for the appropriate operator, and the symbol “–” denotes a logical operation. For example, (□A∧□B)<=>⦶(A∧B).

Third rule: if the logical value of the operational operator is different from the logical value of the operation on variables, the variable operation operator must be adapted to the operational operator. For example: (◊A∧◊B)<=>○(A<=>B).

Fourth rule: in connection with the logical co-valence of the operators of bindingness and impermissibility (|◫A|<=>|⊟B|, |◫B|<=>|⊟A|) as well as permissibility and non-bindingness (|◊A|<=>|⟠B|, |◊B|<=>|⟠A|), the following rules apply when choosing an operational operator:

1) In single-operation formulas, one or another of the co-valent operators can act as the operational operator. For example, the formula ◫A(A≠>B), which can be read as "It’s binding to A." is equivalent to the formula ⊟B(A≠>B), which can be read as "It’s impermissible to B." If a co-valent operational operator is not accompanied by a variable, it should be assumed that its logical value corresponds to the operator with variable A.

2) In formulas involving more than one operation, the operator of the child operation is to be aligned with the operator of the parent operation. For example, regarding the formula F=>F(A∨B), if the antecedent of the competition refers to the variable A, the operational operator of the

disjunction constituting the consequent[1] is also to refer to the variable A, which can be written as FA=>FA(A ∨B). However, if the antecedent refers to variable B, then the operational operator in the consequent must also refer to variable B, which can be written as FB=>FB(A∨B).

Rule five: because the equality, co-bindingness and co-impermissibility operators and their negations refer to both variables, they can be written without adding variables.

Rule six: each logical formula in the CPC can constitute the basis of many deontic formulas. Summary operation tables can be helpful in quickly finding the results of logical operations on operators. For example, one line of such a table, containing the disjunctons of the command A and set of operators B, looks like in the table below:

Table 4

Lp	1	2		3		4		5		6		7		8		9		10		11		12		13		14	
1	$□_A$	$∨□_B$		$∨⧄_A$		$∨⧄_B$		$∨◫_B$		$∨⊟_B$		$∨◊_B$		$∨⟠_B$		∨○		∨⊖		∨⦶		⊕		∨◎		∨⊜	
	1	1	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	1	1
	1	0	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1
	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0	1	1	0	0	1	1
	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	1
		⊜		⊤		◊A ⟠B		⊜		□A		⊤		◊A ⟠B		◊A ⟠B		⊜		□A		⊤		◊A ⟠B		⊤	

On its basis, it can be concluded that the formula with the operational operator "command" is valid in at least two cases: (□A∨⊟B)<=>□(A∨(A≠>B)) and (□A∨⦶)<=>□(A∨(A∧B))

5. Inclusive and exclusive functionality

Logical operations performed on operators reveal that when the variables A and B denote sentences describing mutually exclusive states of affairs – for example, sending and burning a letter – the co-bindingness of these states is not possible. This means that out of four combinations of logical values of the sentences – 11, 10, 01 and 00 – there can be three – 10, 01 and 00. As a result, classical logic becomes reduced logic and the table of logical functions takes the form shown below. Names with a strikethrough indicate those variables and functions whose argument sets are reduced.

Table 5

Lp	1	2	3	4
1	pq	10	01	00
2	antilogy	0	0	0

1 Since the true nature of the implication turned out to be opposition, the name "consequent" should be replaced with the name "opponent" in the future. Footnote JP.

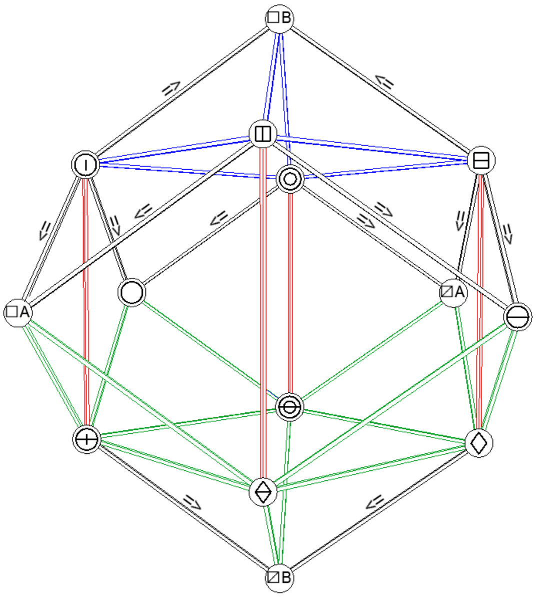
	~~co-bindingness (A∧B)~~			
3	bindingness A, impermissibility B (A≠>B) ~~command A~~	1	0	0
4	impermissibility A, bindingness B (A<≠B) ~~command B~~	0	1	0
5	non-equality (A<≠>B) ~~non-co-impermissibility (A∨B)~~	1	1	0
6	co-impermissibility (A↓B) ~~equality (A<=>B)~~	0	0	1
7	permissibility A, non-bindingness B (A<=B) ~~prohibition B~~	1	0	1
8	non-bindingness A, permissibility B (A=>B) ~~prohibition A~~	0	1	1
9	non-co-bindingness (A↑B) ~~tautology~~	1	1	1

The second degree of reduction is also possible, when the combination of logical values 00 is excluded. This degree refers to sentences describing states of affairs that exclude not only inclusive connection of commands, but also the exclusive connection – for example, such as deciding a football match with penalty kicks.

Consequently, it should be stated that there are two varieties of modal functionality – inclusive, including mutually non-exclusive sentences, and two-stage exclusive, including mutually exclusive sentences. To enable them to be distinguished in the notation, a marking should be introduced to identify the exclusive variety – for example, in the form of a single underscore of the variable B for the first degree and a double underscore B for the second degree.

The geometry of opposition takes a different shape for each type of functionality. In the case of inclusive functionality, the contents of each column of Table 1 create a structure that can be expressed in the form of a rhombic dodecahedron, similar to Pellisier's tetraicosahedron. In the figure below, the vertices of the dodecahedron represent deontic operators. The colors for marking oppositions were adopted in accordance with the standard proposed by Moretti: contrariety – blue, contradiction – red, subcontrariety – green, subalternation – black (Moretti 2009, 80).

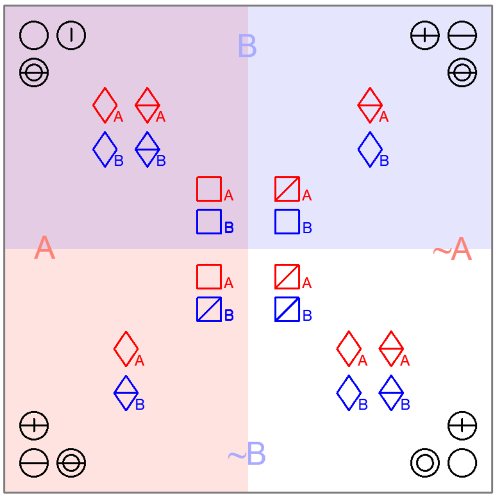
Figure 4

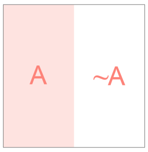
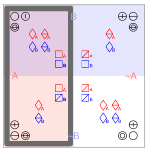
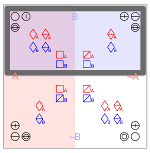
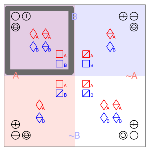
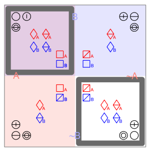
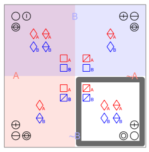
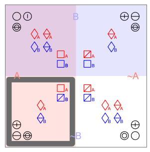
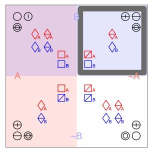
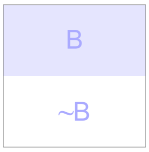
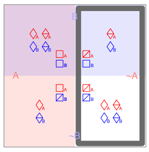
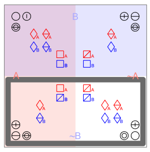
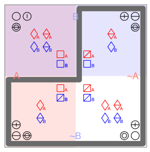
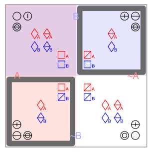
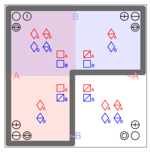
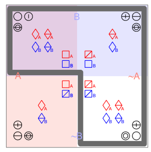
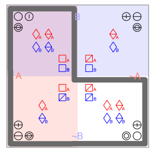


The isness-table method makes it possible to prove that the 36 opposites shown in the figure are tautologies. There are ninety-one oppositions in total between the operators in the dodecahedron, but the twelve of them, which appear between the operators at the pyramid vertices (command A, prohibition A, command B, prohibition B, equality of A and B, non-equality of A and B) as longer diagonals of rhombuses, are undefined relations, that is: there are no logical functions by which these relations could be determined. The remaining 79 oppositions can be called *canonical oppositions*, and the entire structure of the dodecahedron – the *canon of oppositions*.

The oppositions between deontic operators in the examined case can be illustrated using a Venn diagram, identical to the diagram for CPC, or a quadratic diagram, which, thanks to the regularity of divisions, allows for easier remembering of logical operations. The quadratic diagram for inclusive functionality and the table indicating the sets of propositional arguments corresponding to individual operators look like below.

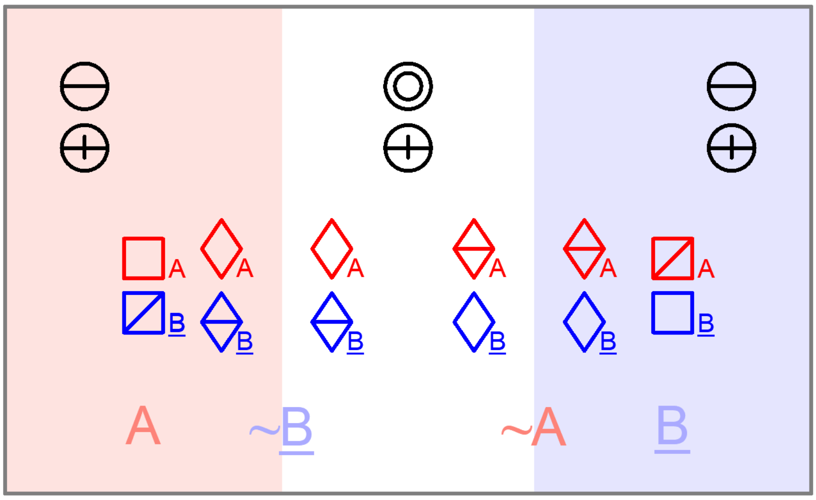
Figure 5

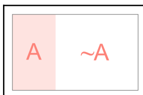
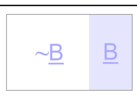
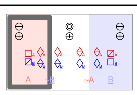
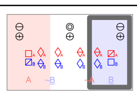
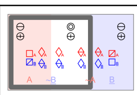
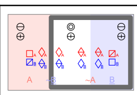
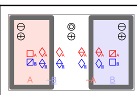
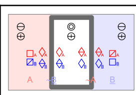
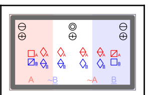


							
A, ~A	□A	□B	⦶	○	◎	◫A, ⊟B	⊟A, ◫B
							
B, ~B	⧄A	⧄B	⊕	⊖	⊜	⟠A, ◊B	◊A, ⟠B

The geometry of the opposition of exclusive functionality does not need to be described in this article. The box diagram for the first-degree exclusive functionality and the table with the sets of modal sentences corresponding to individual operators of the first-degree exclusive functionality look as follows.

Figure 6



								
A, ~A	B, ~B	◫A, ⊟B	⊟A, ◫B	◊A, ⟠B	⟠A, ◊B	⊖	◎	⊕

The second-degree exclusive functionality is reduced to four combinations as in the table below.

Table 6

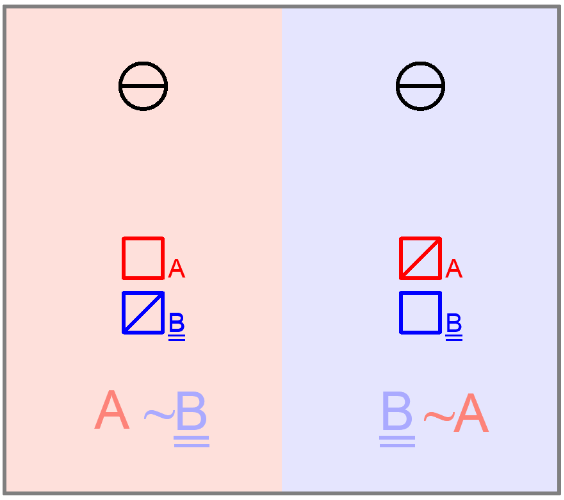
Lp	1	2	3

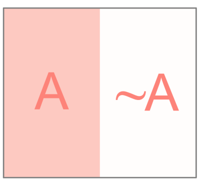
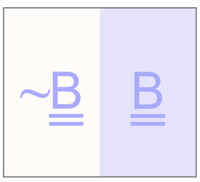
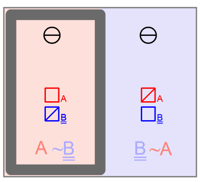
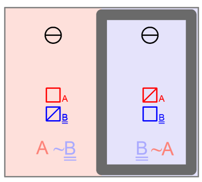
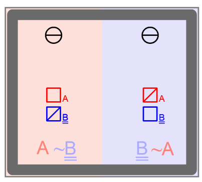
1	pq	10	01
2	antilogy ~~co-bindingness (A∧B)~~ ~~co-impermissibility (A↓B)~~ ~~equality (A<=>B)~~	0	0
3	bindingness A, impermissibility B (A≠>B) ~~command A~~ ~~permissibility A, non-bindingness B (A<=B)~~ ~~prohibition B~~	1	0
4	impermissibility A, bindingness B (A<≠B) ~~command B~~ ~~non-bindingness A, permissibility B (A=>B)~~ ~~prohibition A~~	0	1
5	non-equality (A<≠>B) ~~non-co-impermissibility (A∨B)~~ ~~non-co-bindingness (A↑B)~~ ~~tautology~~	1	1

As in the first-degree exclusive functionality table, the crossed-out names indicate those variables and functions whose argument sets are reduced.

A diagram illustrating the second-degree exclusive functionality and a table illustrating possible sets of modal sentences look like below.

Figure 7



				
A, ~A	B, ~B	◫A, ⊟B	⊟A, ◫B	⊖

This rudimentary functionality refers to commands such as "Save life!" and “Destroy a life!”, “Do good!” and “Do evil!”, “Turn on the light!” and “Turn out the light!” The existence of one state of affairs means the non-existence of the other. As you can see, there is no equilibrium in the mentioned pairs of states. The only case where both variables are covered by one logical function (contravalence) is non-equality.

6. The final procedure for solving Ross's paradox

Having made the above-mentioned findings, we can proceed to the final procedure for solving Ross's paradox. The sentences appearing in this paradox are mutually exclusive sentences, but it is possible that the both states of affairs they describe are impermissible. Therefore, the solution must be carried out using the first-degree exclusive functionality, even though the disjunction in its formula excludes the existence of co-impermissibility.

Based on the formula $p => (p \vee q)$, it is possible to create several dozen tautological deontic formulas with the command operator A in the antecedent, among which the formula $\square A => \square(A \vee B)$ mentioned at the beginning should appear, but now it is immediately visible that there is an error in it: namely, the logical value of the operational operator in the competition consequent is different from the logical value of the operation on variables. To achieve compliance of the formula with the axiom of co-valence, the command operator in the consequent should be replaced with the non-equality operator. The consequent takes the form $\ominus(A \vee B)$. Now it is necessary to adapt the operator in the antecedent and the operation in the consequent to the requirements of first-degree exclusive functionality. Since in this functionality command is reduced to bindingness and non-co-impermissibility to non-equality, the entire formula takes the form $◫ A => \ominus(A <\neq> \underline{\underline{B}})$, which should be read in the language of logic as:

RPD/1: “The letter must be sent; alternatively, the letter should be non-equally sent, if not burned.”

The interpretation presented is strict, but distant from everyday speech. If we want to achieve a sound more similar to plain language, we can decompose the operational operator of non-equality into bindingness operators of variables. The formula $◫ A => (◫ A <\neq> ◫ \underline{B})$ appears, which can be read as:

RPD/2: 'The letter must be sent; alternatively, the letter must be sent, unless it must be burned.”

Checking the reduced tautologicality of formulas using isness tables is presented in the table below:

Table 7

◫A	◫$\underline{B}$	(◫A<≠>◫$\underline{B}$) <=> ⊖(A<≠>$\underline{B}$)	◫A=>(◫A<≠>◫$\underline{B}$) <=> ◫A=>⊖(A<≠>$\underline{B}$
1	0	1	1
0	1	1	1
0	0	0	1

This is the solution to Ross's paradox.

7. Solution to Prior’s paradox

Let us begin the solution of the Prior paradox with the tautology on which Von Wright based his deontic formula. It has the form ~p=>(p=>q). The British logicians Bertrand Russell (1872-1970) and Alfred North Whiteahead (1861-1947) read it as implying that “from a false proposition any proposition follows” (Curley 1975). This reading was probably intended to be a modern version of the *ex falso quodlibet* rule. Von Wright transposed it into the deontic form O~A=>(OA=>OB), which concealed Prior's paradoxical conclusion.

Once the Ross paradox is solved, explaining the Prior paradox does not pose any major difficulties and comes down to reading both the propositional formula ~p=>(p=>q) and the deontic formula O~A=>(OA=>OB) as the competitions.

The propositional formula in the competitive version can be read as:

PPP: “Don't drink coffee; alternatively, drink coffee, but add milk.”

Reading the deontic formula in the form □~A=>□(A=>B) might seem equally clear and simple, but in the light of CMC it turns out to be incorrect for two reasons:

1) the parent competition is not a tautology,

2) the logical value of the child competition is not the same as the logical value of its operation operator.

If the antecedent of the parent competition is to be retained in the formula, the operation operator in the consequent should take on the logical value 0011, 1011, 0111 (tautology 1111 is not a sentence that meets the sense of the formula). The three logical values mentioned correspond respectively to (1) prohibition A, (2) non-bindingness of A and (3) non-co-bindingness of A and B. Combination 1011 is closest to the propositional formula, because the logical value of non-bindingness of A is the same as the logical value of the competition of commands A and B. After inserting the non-bindingness A into the formula, the equivalence appears:

(⧄A=>(☐A=>☐B))<=>(⧄A=>⟠(A=>B))

Consequently, the deontic formula can be read as:

PPD/1: “You are not to drink coffee; alternatively, you are to drink coffee, but you have to drink milk.”

or as:

PPD/2: "You are not drink coffee, but you do not have to drink milk rather than coffee."

As in the case of the solution to the Ross paradox, the formula with operators standing at variables is closer to everyday speech than the formula with an operational operator, but in a good mood it can be considered more sophisticated.

Checking the tautologicality of formulas using isness tables is presented in the table below:

Table 8

☐~A <=> ⧄A	☐ A	☐ B	⟠ <=> (☐A=>☐B)	⧄A=>(☐A=>☐B)	⧄A=>⟠(A=>B)
0	1	1	1	1	1
0	1	0	0	1	1
1	0	1	1	1	1
1	0	0	1	1	1

This is the solution to Prior's paradox.

7. Summary

The resolution of the title paradoxes turned out to be possible thanks to the reading of propositional and deontic formulas based on previous findings, including (1) the real nature of the implication, (2) the real nature of logical values and (3) the geometry of oppositions.

Based on the geometry of oppositions, six deontic operators not previously mentioned in the literature were identified.

Moreover, during the conducted investigations, (1) the Classical Modal Calculus (CMC) was discovered, based on the same principles as the Classical Propositional Calculus (CPC), and (2) a modal diagram, which is a graphical illustration of the CMC.

BIBLIOGRAPHY

1. Hansen J., 2006, *The Paradoxes of Deontic Logic: Alive and Kicking*, in: „Theoria" 72/3, Lund[Oxford] 2006, (pp. 221-232).
2. Hansen J., 2008, *Imperatives and Deontic Logic: On the Semantic Foundations of Deontic Logic*, PhD Dissertation, University of Leipzig.
3. Hansen J., *Imperatives Logic and Its Problems*, in: Gabbay et al. 2013, pp. 137–191.
4. Hedenius I., *Om Ratt och Moral* (*On Law and Morals*), Stockholm 1941.
5. Leibniz G. W., *Elementa Juris Naturalis* [1671]. Comp. *Gottfried Wilhelm Leibniz: Sämtliche Schriften und Briefe*, t. 1. Reihe: Philosophische Schriften, Darmstadt 1930, pp. 465–485.
6. Moretti A., (2009), *The Geometry of Logical Opposition,* PhD Thesis in Logic, University of Neuchâtel, Switzerland.
7. Pociej J./1, *Solving the Paradox of Material Implication – 2024*, https://doi.org/10.6084/m9.figshare.22324282.v3
8. Pociej J./2, *Solving Jörgensen's Dilemma – 2024*, https://doi.org/10.6084/m9.figshare.22329178.v3
9. Ross A., *Imperatives and Logic*, in: „Theoria" 7/1941, pp. 53–71. doi:10.1111/j.1755-2567.1941.tb00034.x

10. SEP (Stanford Encyclopedia of Philosophy), https://plato.stanford.edu/entries/logic-deontic/index.html#ref-78%20Deontic%20logic, access: 03.03.2023.
11. Van Fraassen, *Values and the Heart's Command*, in: „The Journal of Philosophy", 70(1)/ 1973, pp. 5–19. doi:10.2307/2024762
12. Wright G. H. von, *A Note on Deontic Logic and Derived Obligation*, „Mind", 65/1956, pp. 507–509.
13. Wright G.H. von, *Deontic Logic*, in: "Mind", 60(237)/1951, pp. 1–15. doi:10.1093/mind/ LX.237.1
14. Wright, G.H. von, *Norm and Action: A Logical Enquiry*, New York 1963.