The converse implication operator EQT as a tense connective

Abstract

The converse implication operator named EQT arises to study the inequality in the tense of time for Past > Present > Future. EQT is symmetrically bivalent with the 8-bit pattern {1101 1101} as decimal 187. It is shown that: Past in terms of Present is a falsity; Present in terms of Present is a tautology; and Future in terms of Present is a tautology. Derivations are by Peirce NOR and a 2-tuple truth table.

Background

The four commonly used logical connectives as operators at the bit level are AND, IMP, OR, and XOR. The two most commonly used negations are NAND for NOT(AND) and NOR for NOT(OR). The four commonly used operators were developed by Charles S Peirce (1880) using NOR and by Henry M Sheffer (1913) using NAND.

The value and definitions of the 2-tuple as four valued bit code (4VBC) are the bit pairs of 00 `F` for contradiction `F`, 01 `N` for truthity, 10 `C` for falsity, and 00 `T` for tautology (the designated *proof* value). The bit pairs have a left, sinistro, and falsity side, and also a right, dextro, and truthity side in Table 1.

Bit pair	**Left side, sinistro**	**Right side, dextro**	Meaning
01	0 NOT(Falsity)	1 Truthity	`N` Truthity, Non contingency
10	1 Falsity	0 NOT(Truthity)	`C` Falsity, Contingency
00	0 NOT(Falsity)	0 NOT(Truthity)	`F` Contradiction (truthity AND falsity)
11	1 Falsity	1 Truthity	`T` Tautology (truthity OR falsity)

Table 1. Bit pair meanings

Mapping tense with 4VBC

The states of the time continuum are evaluated as relation with the tense of the time segments assigned to bit pairs in Table 2.

10	>	01	>	11
Past	>	Present	>	Future

Table 2. Bit pairs for tenses in time

Past is defined as falsity because it has transpired and is no longer truthity as Present. Future is defined as truthity or falsity, a tautology, because it is undetermined. No tense is assigned to both truthity and falsity at the same time, a contradiction. There is also no number line associated here because the bits {00} as the absence of a variable or lack of a proposition do not exist as a tense, excluded in Table 2.

Three relations are deduced around the fraction of tense / Present as a unity in Table 3.

Past / Present > Present / Present (unity) > Future / Present

Table 3. Inequality of tense based on Present

The fraction of Present / Present is taken as the only fiducial point of unity.

The groups in relation Table 3 are replaced by letters, the division symbol is removed, and the tenses are substituted with bit pairs from 4VBC in Table 2 to make Table 4, where D is Unity.

B	>	D	>	F
10		01		11
01		01		01

Table 4. 4VBC for tense from Table 3

We translate the relations of Table 6 relations into words to obtain an outcome.

B: Past in terms of Present is a falsity because the Past is transpired and is no longer a truthity as is Present.
D: Present in terms of itself is a tautology as being either truthity or a falsity.
F: Future in terms of Present is either truthity or a falsity as a tautology.

We rewrite Table 4 with proposed outcomes in Table 5.

	B	>	D	>	F
p	10		01		11
q	01		01		01
(p?q)	10		11		11

Table 5. Tense relations with results

We search for a symmetrical bivalent operator to produce the results of Table 7. We find the converse implication operator and name it "equate"or EQT (negation NEQT). The bit-wise operation is in Table 8. The decimal numbers are the binary equivalents from our 8-bit canon of 256-operators.

```
  p  0000 1111          p  0000 1111
  q  0101 0101          q  0101 0101
EQT  1010 1111 187    NEQT  0101 0000   68
```

Table 6. Bit-wise operation of the converse implication operator

Therefore using the converse implication operator as EQT, Table 5 can be rewritten in Table 7.

	B	>	D	>	F
p	10		01		11
q	01		01		01
EQT	10		11		11

Table 7. 4VBC for tense defined by EQT

Alternate definition of the converse implication (EQT)

We begin with the AND operator and the equivalence EQV operator for p 0011 and q 0101 in Table 8.

	0011	p		0011
	0101	q		0101
AND	0001		EQV	1001

Table 8. The logical bit operators AND and EQV

The converse implication (EQT) decomposed as (p AND q) EQV (q) equal to (p EQT q) in Table 9.

	0011	p		0011
	0101	q		0101
AND	0001			:
	0101			:
EQV	1011		EQT	1011

Table 9. (p EQT q) shown identical to (p AND q) EQV (q)

Derivation of converse implication (EQT) by NOR

The definition of the operator in Table 9 may be reduced to NOR operations as follows.

NOT(p) = p NOR p; NOT(q) = q NOR q;
p EQT q ≡ ((p AND q) OR (q))
≡ ((((NOT(p)) NOR (NOT(q)))) NOR ((NOT(p)) NOR (NOT(q)))))
NOR (p NOR q))

Lookup table for converse implication (EQT)

For the converse implication (EQT) operator, the results of the operators on the propositions of p and q are tabulated as row major with p as the index to the rows and q as the index to the columns. The headings for the rows and columns arrange the bits of a 2-tuple in the order of <00, 01, 10, 11> for <contradiction, truthity, falsity, tautology> because the bits as binary numbers, with least significant bit to the right, increase in value as <0, 1, 2, 3>, in Table 10 below.

EQT	00	01	10	11
00	**11**	**11**	**11**	**11**
01	**10**	**11**	**10**	**11**
10	**01**	**01**	**11**	**11**
11	**00**	**01**	**10**	**11**

Table 10. Lookup table for the EQT operator

Pattern of converse implication (EQT) in Meth8/VŁ4

We assume the method and apparatus of Meth8/VŁ4 with Tautology as the designated *proof* value, F as contradiction, N as truthity (non-contingency), and C as falsity (contingency). The 16-valued truth table is row-major and horizontal.

LET p, q: p; q; & And; = Equivalent;
% possibility, possibly, for one or some; # necessity, necessarily, for all or every.

(p EQT q): (p AND q) EQV (q) (1.1)
(p&q) = q ; TFTF TFTF TFTF TFTF (1.2)

Eq. 1.2 is *not* tautologous. We attempt to strengthen Eq.1.2 by injecting the existential quantifier in the first antecedent.

(%p AND q) EQV (q) (2.1)
(%p&q) = q ; TTCT TTCT TTCT TTCT (2.2)

Eq. 1.2.2 is *not* tautologous, but approaches tautology more closely than Eq. 1.2.

Evaluation of tense relations with results in Meth8/VŁ4

We evaluate Table 9 as a relational expression, and rewrite it using Table 11.

Definition	Axiom	Symbol	Name	Meaning	2-tuple	Ordinal
1	(p=p)	T	tautology	proof	11	3
2	(p@p)	F	contradiction	absurdum	00	0
3	(%p>#p)	N	non-contingency	truthity	01	1
4	(%p<#p)	C	contingency	falsity	10	2

Table 11. Axioms of Meth8/VŁ4

From Table 7:

	B.1	>	D.1	>	F.1
p	10		01		11
q	01		01		01
EQT	10		11		11

((%p<#p)&(%p>#p))=(%p<#p) ; NNNN NNNN NNNN NNNN (B.2)
((%p>#p)&(%p>#p))=(p=p) ; NNNN NNNN NNNN NNNN (D.2)
((p=p)&(%p>#p))=(p=p) ; NNNN NNNN NNNN NNNN (F.2)

We rewrite the expression as: B.1 > D.1 > F.1 (3.1)
(((((%p<#p)&(%p>#p))=(%p<#p)) > (((%p>#p)&(%p>#p))=(p=p)))=(p=p))
> (((p=p)&(%p>#p))=(p=p)) ; NNNN NNNN NNNN NNNN (3.2)

Eqs. B.2, D.2, and F.2 as rendered are *not* tautologous, but are respectively truthity as non-contingent.

This means that the converse implication (EQT) as derived implies that tense, and hence time, is not a tautology. Therefore time can only be assumed as non-contingent and truthity, and not as a theorem.