4.9. Conditional and Biconditional Languages: Expressive Adequacy

We earlier explored issues of expressive power and expressive adequacy for the Chapter Three formal language {~, ∧, ∨}, and various of its 'sub-languages'. But with the advent of the arrow and biconditional sign ("bicon") in the Chapter Four language, the issue of expressive adequacy rises again.

The full Chapter Four language {~, ∧, ∨, →, ↔} is *bound* to be **expressively adequate** – capable of supplying a sentence to match any given truth table. For we established already that the Chapter Three language {~, ∧, ∨} is expressively adequate. But every {~, ∧, ∨} sentence qualifies as a {~, ∧, ∨, →, ↔} sentence. So any possible truth table will be matched by some {~, ∧, ∨, →, ↔} sentence – making the {~, ∧, ∨, →, ↔} language expressively adequate.[1]

More interesting is the question whether there are any expressively adequate *sub-languages* of {~, ∧, ∨, →, ↔} which feature arrow or bicon.

In fact there are. And the simplest of these is the {~, →} language.

To prove {~, →} expressively adequate we use the same strategy applied earlier to the {~, ∧} language. Recall that, having established that {~, ∧, ∨} is expressively adequate, we showed that {~, ∧} sentences can generate any truth table which {~, ∧, ∨} sentences can –making {~, ∧} adequate as well. Since {~, ∧} is the same as {~, ∧, ∨} but for lack of vel, the trick was to find a {~, ∧} form semantically equivalent to a disjunction. Finding such a form – "~(~● ∧ ~▲)" in place of "(● ∨ ▲)" – settled that {~, ∧} could build any truth table which {~, ∧, ∨} could. And a similar strategy established the adequacy of the {~, ∨} language.

---

[1] In general: adding further connectives to a formal language can only increase its expressive power – the set of truth tables covered by the sentences of that language.

Likewise, if we show that {~, →} is expressively equivalent to an expressively adequate language, we settle that {~, →} is itself expressively adequate.

We achieve that end by constructing a {~, →} sentence form semantically equivalent to the conjunction. "~(● → ~▲)," will always take the same truth table as "(● ∧ ▲)".

●	▲	~▲	(● → ~▲)	~(● → ~▲)	(● ∧ ▲)
1	1	0	0	**1**	**1**
1	0	1	1	**0**	**0**
0	1	0	1	**0**	**0**
0	0	1	1	**0**	**0**

{~, →} sentences can therefore cover all of the truth tables which {~, ∧} sentences can. But since {~, ∧} is expressively adequate, its sentences cover *all* possible truth tables. So {~, →} sentences do as well –establishing that **{~, →} is expressively adequate**.

And for the reasons rehearsed above, any larger language containing arrow and tilde will be expressively adequate as well – for example, {~, →, ↔}.

Indeed, any *adequate* sub-language of {~, ∧, ∨, →, ↔} must contain a tilde, and either a wedge, vel, or arrow. So these are the adequate sub-languages of {~, ∧, ∨, →, ↔}.

{~, ∧, ∨, →}	{~, ∧, ∨}	{~, ∧}
{~, ∧, ∨, ↔}	{~, ∧, →}	{~, ∨}
{~, ∧, →, ↔}	{~, ∨, →}	{~, →}
{~, ∨, →, ↔}	{~, ∧, ↔}	
	{~, ∨, ↔}	
	{~, →, ↔}	

We show that all remaining sub-languages of {~, ∧, ∨, →, ↔} are expressively **inadequate**.

Of the single-connective languages, {~}, {∧}, and {∨} were proven inadequate in the previous chapter. And the argument which applied to {∧} and {∨} works as well for {→} and {↔}. Recall that any formal sentence built from a wedge or vel (along with sentence letters and parentheses) will be **true in the first valuation**. But that holds as well for sentences built from an arrow or bicon.

●	▲	(● → ▲)	(● ↔ ▲)
1	**1**	**1**	**1**
1	0	0	0
0	1	1	0
0	0	1	1

So no {→} or {↔} sentence will match the truth table for a negation, which is false in the first valuation. Indeed, none of the following formal languages has a sentence matching a negation truth table; so **all are expressively inadequate**.

{∧, ∨, →}	{∧, ∨}	{∧}
{∧, ∨, ↔}	{∧, →}	{∨}
{∧, →, ↔}	{∨, →}	{→}
{∨, →, ↔}	{∧, ↔}	{↔}
	{∨, ↔}	
	{→, ↔}	

The only remaining sub-language is {~, ↔}. But we observe a remarkable feature of tildes and bicons in combination: if a biconditional or negation of one has more than one tilde, it is logically equivalent to either a biconditional with no tildes, or a biconditional with a single tilde.

For instance, "~(P ↔ Q)," "(~P ↔ Q)," and "(P ↔ ~Q)" all take the same truth table. So the result of adding further tildes to any of these sentences will yield a sentence that has the same truth table as either "(P ↔ Q)" or "(P ↔ ~Q)".

P	**Q**	**~P**	**~Q**	**(P ↔ Q)**	**~(P ↔ Q)**	**(~P ↔ Q)**	**(P ↔ ~Q)**
1	1	0	0	1	0	0	0
1	0	0	1	0	1	1	1
0	1	1	0	0	1	1	1
0	0	1	1	1	0	0	0

For {~, ↔} sentences built from "P" and/or "Q," the only further tables picked out are the contradiction truth table (taken by, e.g., "(P ↔ ~P)"), and the tautology truth table (taken by, , e.g., "(P ↔ P)"). So all "P" and "Q" sentences in the {~, ↔} language take one of these four truth tables, or the truth table for "P" or "Q" or their negations – eight truth tables in all.

P	**Q**	**~P**	**(P ↔ Q)**	**(P ↔ ~Q)**	**(P ↔ P)**	**(P ↔ ~P)**
1	1	0	1	0	1	0
1	0	0	0	1	1	0
0	1	1	0	1	1	0
0	0	1	1	0	1	0

Note that in this language, every sentence is true in an **even number of valuations**. That feature holds for all sentence letters, and negations and biconditionals of sentence letters; and will continue to hold for any larger biconditional or negation in this language. So that feature holds for all {~, ↔} sentences.

But a number of familiar truth tables don't have an even number of true valuations – e.g., the truth tables for "(P ∧ Q)," "(P ∨ Q)," and "(P → Q)". Offering no sentences which takes such a truth table, **the {~, ↔} language is expressively inadequate**. (Indeed, {~, ↔} is the *only* 2-connective sub-language of {~, ∧, ∨, →, ↔} which contains the tilde, but is still expressively inadequate.)

Summary

The Chapter Four Language and Its Sub-Languages:
Expressive Power