Alternate connectives / Adequate sets

So far: $\neg, \wedge, \vee, \to, \leftrightarrow$

Also $|$ - Nand, $\downarrow$ - Nor $\Big\}$ Adequate by themselves: $\{|\}$, $\{\downarrow\}$ are adequate

$\{\neg, \wedge\}$, $\{\neg, \vee\}$, $\{\neg, \to\}$ are adequate

$\{\wedge, \vee, \to, \leftrightarrow\}$ is not adequate.

$\{\neg, \wedge, \leftrightarrow\}$ - Yes. Since $\{\neg, \wedge, \leftrightarrow\} \supseteq \{\neg, \wedge\}$

Thm $\{\neg, \leftrightarrow\}$ is not adequate. **Example:** $(p_1 \wedge p_2)$ is not definable by a $\{\neg, \leftrightarrow\}$-formula

Pf Explore what can be expressed.

p_1	p_2	$\neg p_1$	$\neg p_2$	$p_1 \leftrightarrow p_2$	$\neg(p_1 \leftrightarrow p_2)$	$p_1 \leftrightarrow p_1$	$\neg(p_1 \leftrightarrow p_1)$
T	T	F	F	T	F	T	F
T	F	F	T	F	T	T	F
F	T	T	F	F	T	T	F
F	F	T	T	T	F	T	F

$(A \leftrightarrow \neg B) \models\!\!\!= \neg(A \leftrightarrow B)$

Qed #1

Pf #2 $\leftrightarrow$ - commutative and associative

And $(\neg p \leftrightarrow q) \models\!\!\dashv \neg(p \leftrightarrow q)$

So and $\{\neg, \leftrightarrow\}$-formula on p_1 and p_2 is equivalent to

$$\neg\neg \cdots \neg(p_1 \leftrightarrow p_1 \leftrightarrow p_1 \leftrightarrow \cdots p_1 \leftrightarrow p_2 \leftrightarrow p_2 \leftrightarrow \cdots p_2)$$

$(p_1 \leftrightarrow p_1 \leftrightarrow A) \models\!\!\dashv A \qquad \neg\neg A \models\!\!\dashv A$

So wlog at most $\neg$ sign.

and at most one p_1 and at most one p_2

Now back to previous page.

More connectives

$\perp$ - constant false nullary 0-ary \perp

$\top$ - constant true " \top

Parity $\oplus$ - exclusive or \oplus

p_1	p_2	$p_1 \oplus p_2$
T	T	F
T	F	T
F	T	T
F	F	F

$p \oplus q \models\!\dashv \neg(p \leftrightarrow q)$

Same as addition mod 2

Identify T with 1 and F with 0

then $\varphi(p \oplus q) = \varphi(p) + \varphi(p) \bmod 2$.

C, C++, Java a ^ b - bitwise xor

Case connective: Switch, Select, If-Then-Else

$$\mathrm{Case}(p_1, p_2, p_3) = \begin{cases} \text{value of } p_2 & \text{if } p_1 \\ \text{value of } p_3 & \text{if } p_1 \text{ is false} \end{cases}$$

$$\varphi(\mathrm{Case}(p_1, p_2, p_3)) = \begin{cases} \varphi(p_2) & \text{if } \varphi(p_1) = T \\ \varphi(p_3) & \text{if } \varphi(p_1) = F \end{cases}$$

3-ary

Majority $\mathrm{Maj}^3(p_1, p_2, p_3)$

$\varphi(\mathrm{Maj}^3(p_1, p_2, p_3)) = T$ iff at least 2 of $\varphi(p_1)$, $\varphi(p_2)$ $\varphi(p_3) = T$

k-ary $\varphi(\mathrm{Maj}^k(p_1, \ldots, p_k)) = T$ if " " $k/2$ of $\varphi(p_i)$'s $= T$.

k-ary $\vee$: $\varphi(\vee^k(p_1, \ldots, p_k)) = T$ if $\varphi(p_i) = T$ for some i

k-ary $\wedge$ $\varphi(\wedge^k(p_1 \ldots p_k)) = T$ if " ... " all i

k-ary $\oplus$ $\varphi(\oplus^k(p_1 \ldots p_k)) = T$ if " " an odd number of i's.

Substitution I Substituting a tautologically equivalent subformula

Theorem: Let A be a formula, let B be a subformula of A. Let $C \models\!\dashv B$. Let A^* be obtained from A by replacing the subformula B with C.

Then $A^* \models\!\dashv A$

Example: $p \to (q \to \neg r)$ is A $(q \to \neg r)$ is B $\neg(q \wedge r)$ is C

$$p \to \underbrace{(q \to \neg r)} \models\!\dashv p \to \underbrace{\neg(q \wedge r)}$$

Substitution II Substitute for variables in a tautology

Theorem Let A be a tautology. Let $B_1, .., B_k$ be formulas. Let $p_{i_1}, .. p_{i_k}$ be variables.

Let $A(B_1, .., B_k / p_{i_1}, .., p_{i_k})$ be ~~the~~ result of replacing every p_{i_j} in A with B_j

Then $A(B_1, .., B_k / p_{i_1}, .., p_{i_k})$ is a tautology.

Example $\models p_1 \to (p_2 \to p_1)$. Therefore $\models B \to (C \to B)$ for any B, C.

We're going to formally define proofs

Proof System $\underline{PL}$ (Hodel's book §3.5, System "L") Hilbert-style (David Hilbert)

$\underline{PL}$ - Rule of inference $\underline{\text{modus ponens}}$ $\dfrac{A \quad A \to B}{B}$

Step-by-step reasoning.

$\underline{PL\text{-axioms}}$: $A \to (B \to A)$

$$(A \to (B \to C)) \to ((A \to B) \to (A \to C))$$

$$\neg A \to (A \to B)$$

$$(\neg A \to A) \to A$$

Proofsystems:

Proof - string of symbols ("expression")

- obeys syntactic rules"

Desirable Properties of a Proof System

	PL	truth Tables TT
(1) Algorithmic - There be an algorithm to recognize valid proofs and what is being proved	✓	✓
(2) Soundness - only tautologies are proved. More generally $\Gamma \vDash A$ is proved only if it is true	✓	✓
(3) Completeness: all tautologies have proof. If $\Gamma \vDash A$, then there is proof of this (Even for infinite sets Γ.)	✓	✓
(4) User-friendly (Human-centric)	✓~	✗
(a) Simulate human reasoning	✓	✗
(b) Step-by-step reasoning	✓	✗
(c) Proofs should humanly understand.	~	?
(5) Elegant	✓	✗
(6) Efficient Proof Search (by people or computers)	? Open Question	✗

"Feasible" - not exponential time

feasible ≈ polynomial time