CHANGING RULES

Change $\frac{P \& Q}{P}$: sound, not complete.
proof for not complete:
(counterex) $P \& Q \; \therefore Q$ → $P \& Q$, $\sim Q$ / P / Q / [illegible] using new rule: $P \& Q$, $\sim Q$ / P / ✓ / ↑ [illegible] counterex

Change $\frac{P \vee Q}{P}$: complete, not sound
proof for not sound:
(counterex) $P \vee Q \; \therefore P$ → $P \vee Q$, $\sim P$ / P ✗, Q ✓ using new rule: $P \vee Q$, $\sim P$ / P / ✗ / ↑ [illegible]

TEST FOR CONSISTENCY: [illegible] if [illegible], say NO

TEST FOR LOGICALLY EQUIVALENT: ($A \equiv$)

ex) Are $(A \& B) \supset C$ and $A \supset (B \supset C)$ l.e.?

$(A \& B) \supset C$ ✓
$\sim (A \supset (B \supset C))$ ✓
$\sim (A \& B)$ / C
$\sim A$, $\sim B$ / A
A, A / $\sim (B \supset C)$
$\sim (B \supset C)$ ✗, $\sim (B \supset C)$ / B
B / $\sim C$ ✗
$\sim C$ ✗

$\sim ((A \& B) \supset C)$
$A \supset (B \supset C)$ ✓
$\sim A$ / $(B \supset C)$
$(A \& B)$ / $\sim B$, C
$\sim C$ / $(A \& B)$, $(A \& B)$
A / $\sim C$, $\sim C$
B / A ✗
✗ / B ✗

∴ [illegible] logically equiv.

TEST FOR LOGICALLY TRUE: $\sim A \supset (B \& C)$ [illegible]
$\sim (A \supset (B \& C))$
A
$\sim (B \& C)$
$\sim B$ / $\sim C$
∴ [illegible] not logically true

ex) $A \supset (A \vee C)$ [illegible]
$\sim (A \supset (A \vee C))$
A
$\sim (A \vee C)$
$\sim A$
$\sim C$
✗ ∴ logically true

SOUND : gives no false positives (if tree says yes, then yes is correct)

proof: Soundness of the tree test for validity in SL
Lemma #1: If a given row of the truth table makes a certain sentence true and we apply the tree rule to that sentence, then that same row will make at least one of the resulting branches true
- proof of the lemma:
  - atomics and neg atoms : no tree rule
  - conjunction $P \& Q$: Every truth table row that makes $P \& Q$ true, also makes P and Q true
  - neg conjunction $\sim(P \& Q)$: Every truth table row that makes $\sim(P \& Q)$ true, also makes at least one of $\sim P$, $\sim Q$ true
  - etc. $(P \vee Q)$, $\sim(P \vee Q)$, $(P \supset Q)$, $\sim(P \supset Q)$, $(P \equiv Q)$, $\sim(P \equiv Q)$

Lemma #2: If some row of truth table makes all the sentences on a certain branch true, then after applying any tree rule to a sentence on that branch at least one of the resulting branches remains open

COMPLETENESS : gets all the correct positives (if yes is correct, then the tree will say yes) ; does not hallucinate counterexamples

proof: completeness of the tree test for validity in SL: if tree does not close, then the arg is not valid
Lemma: going from the bottom to the top of a finished branch, assign true to each atomic sentence and false to each negated atomic. You get a row of the truth table that makes every sentence on that branch true.
- proof of the lemma:
  - atomics and neg : cannot be made false (b/c that's what we said above)
  - Conj $P \& Q$: cannot be the first sentence to be false since below we have already made P true and Q true
  - neg conjunction : cannot be the first sentence to be false because we have already made $\sim P$ true or $\sim Q$ true
  - etc.

[illegible]

EFFECTIVENESS : always gives an answer
Theorem: If a tree for sentences has no bi-conditionals and has exactly [illegible] binary connectives and
n negations, then the tree will be finished with at most [illegible] steps
i) [illegible]
ii) [illegible]
iii) [illegible]

Tree Rules

ex) $\exists x Px$ ← someone is a poet
Pa ← Alan is a poet

ex) a, b, c ← constants that appeared above
$\forall x Px$ ← cannot be checked off, may be constants later on
$\left.\begin{array}{l} Pa \\ Pb \\ Pc \end{array}\right\}$ for every constant on the branch

Strategy for seeing if the branch will be infinite

1) Sentences on the branch are checked
2) $\forall x \exists y$

Open branch is "finished" when:
1) Every $\wedge, \vee, \supset, \equiv, \sim\wedge, \sim\vee, \sim\supset, \sim\equiv, \exists, \sim\exists, \sim\forall, \sim\sim$ has been checked
2) Every $\forall$ on the branch has been applied to every constant on the branch

Computer:
Sound - yes
Complete: yes
effective: no

Modify: Stop after N steps and say "no"
Sound: yes
Complete: no
effective: yes

Modify: Stop after N steps and say "yes"
Sound: no
Complete: yes
effective: yes

Universal vs Particular (same)
Affirmative vs Negative

ex) Every Poet is a Dancer
Every Dancer is a Musician
∴ Every Poet is a Musician

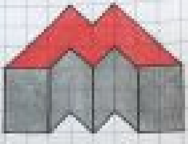
$\forall x(Px \supset Dx)$
$\forall x(Dx \supset Mx)$
$\therefore \forall x(Px \supset Mx)$

$\forall x(Px \supset Dx)$
$\forall x(Dx \supset Mx)$
$\sim(\forall x(Px \supset Mx))$

$\exists x \sim(Px \supset Mx)$
Pa
$\sim Ma$
$Pa \supset Da$

$\sim Pa$ Da
X $Da \supset Ma$

$\sim Da$ Ma ∴ Valid
X   X



ex) Every Poet is a Dancer
Every Dancer is a Musician
∴ Every Musician is a Poet

$\forall x(Px \supset Dx)$
$\forall x(Dx \supset Mx)$
$\therefore \forall x(Mx \supset Px)$

$\forall x(Px \supset Dx)$
$\forall x(Dx \supset Mx)$
$\sim(\forall x(Mx \supset Px))$

$\exists x \sim(Mx \supset Px)$
Ma
$\sim Pa$
$Pa \supset Da$

$\sim Pa$ Da
$Da \supset Ma$ $Da \supset Ma$

$\sim Da$ Ma $\sim Da$ Ma ∴ Not valid
X   ↑

counterexample:
UD = {a}
a : a
P : ∅
D : {a}
M : {a}

- Universal vs. Particular (some)
- Affirmative vs. Negative

ex) Every Poet is a Dancer
Every Dancer is a Musician
∴ Every Poet is a Musician

$\forall x(P_x \supset D_x)$
$\forall x(D_x \supset M_x)$
$\therefore \forall x(P_x \supset M_x)$

$\forall x(P_x \supset D_x)$
$\forall x(D_x \supset M_x)$
$\sim(\forall x(P_x \supset M_x))$

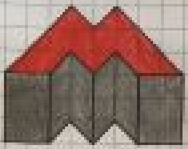
$\exists x \sim(P_x \supset M_a)$
P_a
$\sim M_a$
$P_a \supset D_a$

$\sim P_a$ ✗ | D_a

$D_a \supset M_a$

$\sim D_a$ ✗ | M_a ✗

Valid



---

ex) Every Poet is a Dancer
Every Dancer is a Musician
∴ Every Musician is a Poet

$\forall x(P_x \supset D_x)$
$\forall x(D_x \supset M_x)$
$\forall x(M_x \supset P_x)$

$\forall x(P_x \supset D_x)$
$\forall x(D_x \supset M_x)$
$\sim(\forall x(M_x \supset P_x))$

$\exists x \sim(M_a \supset P_a)$
M_a
$\sim P_a$
$P_a \supset D_a$

$\sim P_a$ | D_a

$D_a \supset M_a$ | $D_a \supset M_a$

$\sim D_a$ | M_a | $\sim D_a$ ✗ | M_a ↑

∴ not valid

counterexample:
UD: {a}
a: a
P: ∅
D: {a}
M: {a}

ex) Some Sailors are Carefree and unlucky
Some people are lucky
∴ Some people are not sailors

$\exists x\,(Sx \,\&\, Cx \,\&\, Ux)$
$\exists x\,(\sim Ux)$
$\therefore \exists x\,(\sim Sx)$

Countermodel:
U.D. = {a, b}
a: [illegible]
b: [illegible]
S: {a, [illegible]}
C: {a}
U: {a}

$\exists x\,(Sx \,\&\, Cx \,\&\, Ux)$
$\exists x\,(\sim Ux)$
$\underline{\sim \exists x\,(\sim Sx)}$
$\forall x \sim(\sim Sx)$
Sa
Ca
Ua
$\sim Ub$
Sb

---

ex) $\forall x\,(Dx \supset \sim Px)$
$\exists x\, Px$
$\therefore \exists x \sim Dx$

$\forall x\,(Dx \supset \sim Px)$
$\exists x\, Px$
$\underline{\sim(\exists x \sim Dx)}$
Pa
$\forall x \sim\sim Dx$
$\sim\sim Da$
$Da \supset \sim Pa$

$\sim Da$ — X $\sim Pa$ — X

ex) $\forall x\,(Px \supset Dx)$
$\forall x\,(Dx \supset Mx)$
$\underline{\sim \exists x\, Mx}$
$\forall x \sim Mx$
$\sim Ma$
$Pa \supset Da$

$\sim Pa$ Da
$Da \supset Ma$ $Da \supset Ma$
$\sim Da$ Ma — X $\sim Da$ — X Ma — X

(x) $\forall x(Gx \supset Kxs)$
Gs
$\therefore Kss$

$\forall x(Gx \supset Kxs)$
Gs
$\sim(Kss)$

Gs
$Gx \supset Kxs$
$Gs \supset Kss$

$\sim Gs$ $\quad$ Kss
$\times$ $\quad$ $\sim Kss$
$\times$

$\exists x Gx$
$\exists x Px$
$\therefore \exists x(Gx \& Px)$

$\exists x Gx$
$\exists x Px$
$\sim(\exists x(Gx \& Px))$

Pa
Gb
$\forall x \sim(Gx \& Px)$
$\sim(Gx \& Px)$
$\sim(Ga \& Pa)$

$\sim Ga$ $\quad$ $\sim Pa$
$\sim(Gb \& Pb)$ $\quad$ $\times$

$\sim Gb$ $\quad$ $\sim Pb$
$\times$ $\quad$ $\uparrow$