

6.0341 FALL 2020
RECITATION 4: RBS

- DEFINITIONS
- FORWARD CHAINING
- BACKWARD CHAINING

DEFINITIONS

RULE: IF (antecedent)

THEN (consequent)

* DELETE . [*optional]

ASSERTIONS:

statements we hold to be true

(A₀, A₁, A₂ ...)

How DOES IT WORK?

FORWARD CHAINING

- Start with assertions
- Apply rules until finished

BACKWARD CHAINING

- Start with a hypothesis
- Build a goal tree

FORWARD CHAINING

STEPS:

matches = []

for each rule in all-rules:

 for each assertion in all-assertions:

 if match (rule.antecedent, assertion):

 rule_match = rule.update (assertion)

 matches.append (rule-match)

for each rule-match in matches:

 if rule-match.assertion NOT in all-assertions:

 rule_match.fire () # restart or break

Problem 1: Rule Systems (50 points)

Engaged in a debate regarding reverse psychology and self-fulfilling prophecy, the 6.034 TAs decide to write a Rule-Based system to determine, once and for all, whether Neo should be able to determine if he is The One in the original film 'The Matrix.'

Rules:

P0	IF (OR '(?x) is a noob', '(?x) is disenchanted') THEN '(?x) visits The Oracle'
P1	IF '(?x) visits The Oracle' THEN '(?x) picks The One'
P2	IF (AND '(?x) is a noob', '(?x) visits The Oracle') THEN 'The Oracle says (?x) is not The One', '(?x) is disenchanted'
P3	IF (AND (OR '(?x) is disenchanted', '(?x) is a noob'), 'The Oracle says (?x) is not The One') THEN '(?x) practices Kung Fu'
P4	IF (AND '(?x) is disenchanted', '(?x) loves (?y)') THEN '(?x) proclaims love for (?y)')
P5	IF (AND 'The Oracle says (?x) is not The One', '(?y) proclaims love for (?x)', '(?y) picks The One') THEN '(?x) is The One'

Assertions:

- A0: (Neo is a noob)
- A1: (Trinity is disenchanted)
- A2: (Trinity loves Neo)
- A3: (Trinity proclaims love for Neo)

Iteration	Matches	Fire	New Assertion
1	P0, P4	P0($x=Ne$)	A4: Neo visits the Oracle
2	P0, P1, P2P4	P0($x=\text{Trinity}$)	A5: Trinity visits the Oracle
3	P0, P1, P2, P4	P1($x=Ne$)	A6: Neo picks the one
4	P0, P1, P2, P4	P1($x=\text{Tr.}$)	A7: Trinity picks the one
5	P0, P1, P2, P4	P2($x=Ne$)	A8: The Oracle says Neo is not the one A9: Neo is disenchanted
6	P0 - P5	P3($x=Ne$)	A10: Neo practices kung-fu
7	P0 - P5	P5($\begin{smallmatrix}x=Ne \\ y=\text{Tr.}\end{smallmatrix}$)	A11: Neo is the One
8	...		

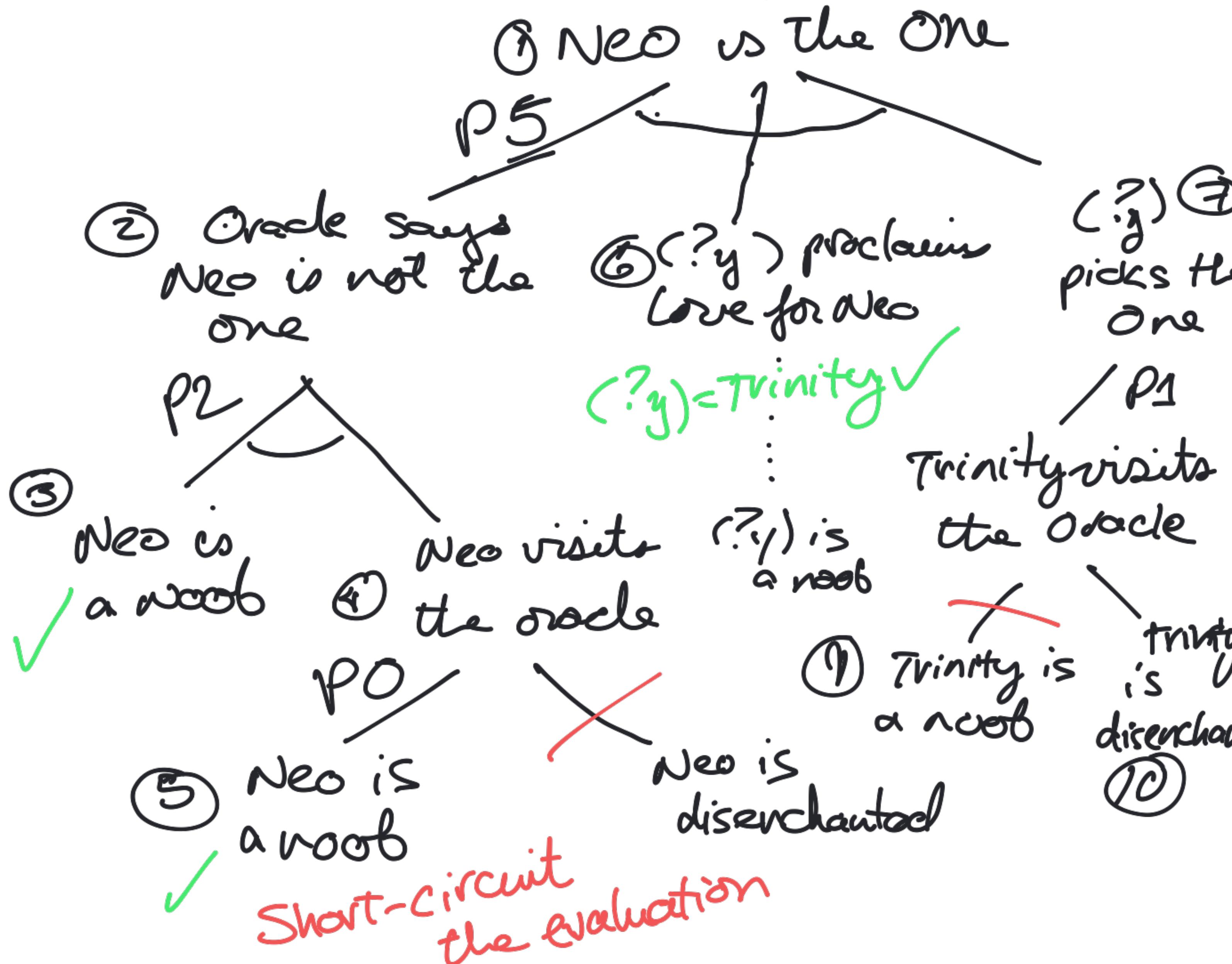
BACKWARD CHAINING

STEPS :

- 1) Check whether the **hypothesis** is in our list of assertions
→ IF so, we're DONE!
- 2) Find all the rules with consequents that match the hypothesis
- 3) Build the tree of found antecedents
- 4) Depth-first search with backtracking if we get stuck

BACKWARD CHAINING

Root



Checked

- 1) Neo is the one
- 2) Oracle says Neo is not the One
- 3) Neo is a noob
- 4) Neo visits the Oracle
- 5) Neo is a noob
- 6) (?y) = Trinity
- 7) Trinity picks the One
- 8) Trinity visits the Oracle
- 9) **SHORT CIRCUIT**
- 10) Trinity is disenchanted

A WORD ABOUT NOT(·)

P0: IF (AND ((?x) has chocolate ,
NOT ((?x) does not like chocolate)),
THEN ((?x) is happy)

A0: John has chocolate

A1: John despises chocolate

Want to know: Is John happy?

