Inference in FOPL

28/01/2025

Koustav Rudra

Inference Rules

- Universal Elimination
  - $\forall_x$Likes(x, flower)
  - Substituting x by Akash {x/Akash}
    - Likes(Akash, flower)

- The substitution should be done by a constant term

Inference Rules

- Existential elimination (Skolemization)
  - $\exists_x$Likes(x, flower) ➔Likes(Person, flower)
    - as long as Person is not in the knowledge base

- This method of finding out a particular constant that satisfies the predicate there exists x that likes flower is known as Skolemization.

- Existential introduction
  - Likes(John, flower)
  - Can be written as
    - $\exists_x$Likes(x, flower)

Example1: Reasoning in FOL

- The law says that it is a crime for a Gaul to sell potion formulas to hostile nations.

- The country Rome, an enemy of Gaul, has acquired some potion formulas, and all of its formulas were sold to it by Druid Traitorix.

- Traitorix is a Gaul

- Is Traitorix a criminal?

Example1: Reasoning in FOL

- The law says that it is a crime for a Gaul to sell potion formulas to hostile nations.
- Gaul(x): x is a Gaul
- Hostile(z): z is hostile nation
- Potion(y): y is a potion
- Criminal(x): x is criminal
- Sells(x,y,z): x sells y to z
- Owns(x,y): x owns y

- $\forall_x \forall_y \forall_z$ Gaul(x) ∧ Potion(y) ∧ Sells(x,y,z) ∧ hostile(z) → Criminal(x)

Example1: Reasoning in FOL

- The country Rome, an enemy of Gaul, has acquired some potion formulas, and all of its formulas were sold to it by Druid Traitorix.
- Gaul(x): x is a Gaul
- Hostile(z): z is hostile nation
- Potion(y): y is a potion
- Criminal(x): x is criminal
- Sells(x,y,z): x sells y to z
- Owns(x,y): x owns y

- Hostile(Rome)
- $\exists_y$ Potion(y) $\wedge$ Owns(Rome,y)
- $\forall_y$ Potion(y) $\wedge$ Owns(Rome,y) $\rightarrow$ Sells(Traitorix, y, Rome)

Example1: Reasoning in FOL

- Traitorix is a Gaul

- Gaul(x): x is a Gaul
- Hostile(z): z is hostile nation
- Potion(y): y is a potion
- Criminal(x): x is criminal
- Sells(x,y,z): x sells y to z
- Owns(x,y): x owns y

- Gaul(Traitorix)
- ?- Criminal(Traitorx) [We have to deduce it]

Example1: Reasoning in FOL

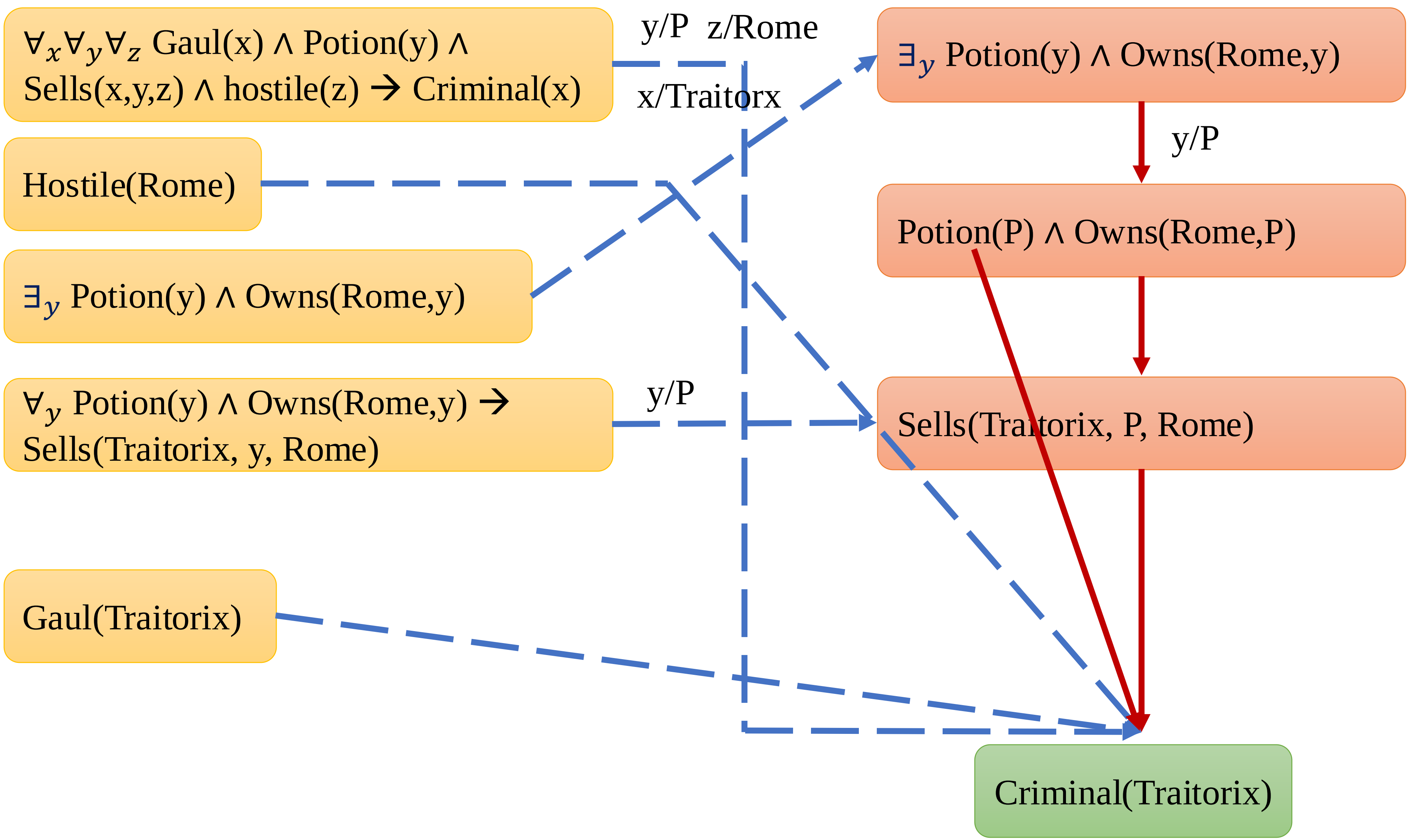
- $\forall_x \forall_y \forall_z$ Gaul(x) ∧ Potion(y) ∧ Sells(x,y,z) ∧ hostile(z) → Criminal(x)
- Hostile(Rome)
- $\exists_y$ Potion(y) ∧ Owns(Rome,y)
- $\forall_y$ Potion(y) ∧ Owns(Rome,y) → Sells(Traitorix, y, Rome)
- Gaul(Traitorix)

- ?- Criminal(Traitorx)

Forward Chaining → Started from existing facts and rules in knowledge base and reach the goal

Backward Chaining → Started from the goal and use existing facts and rules in knowledge base to prove goal

Reasoning in FOL: Forward Chaining

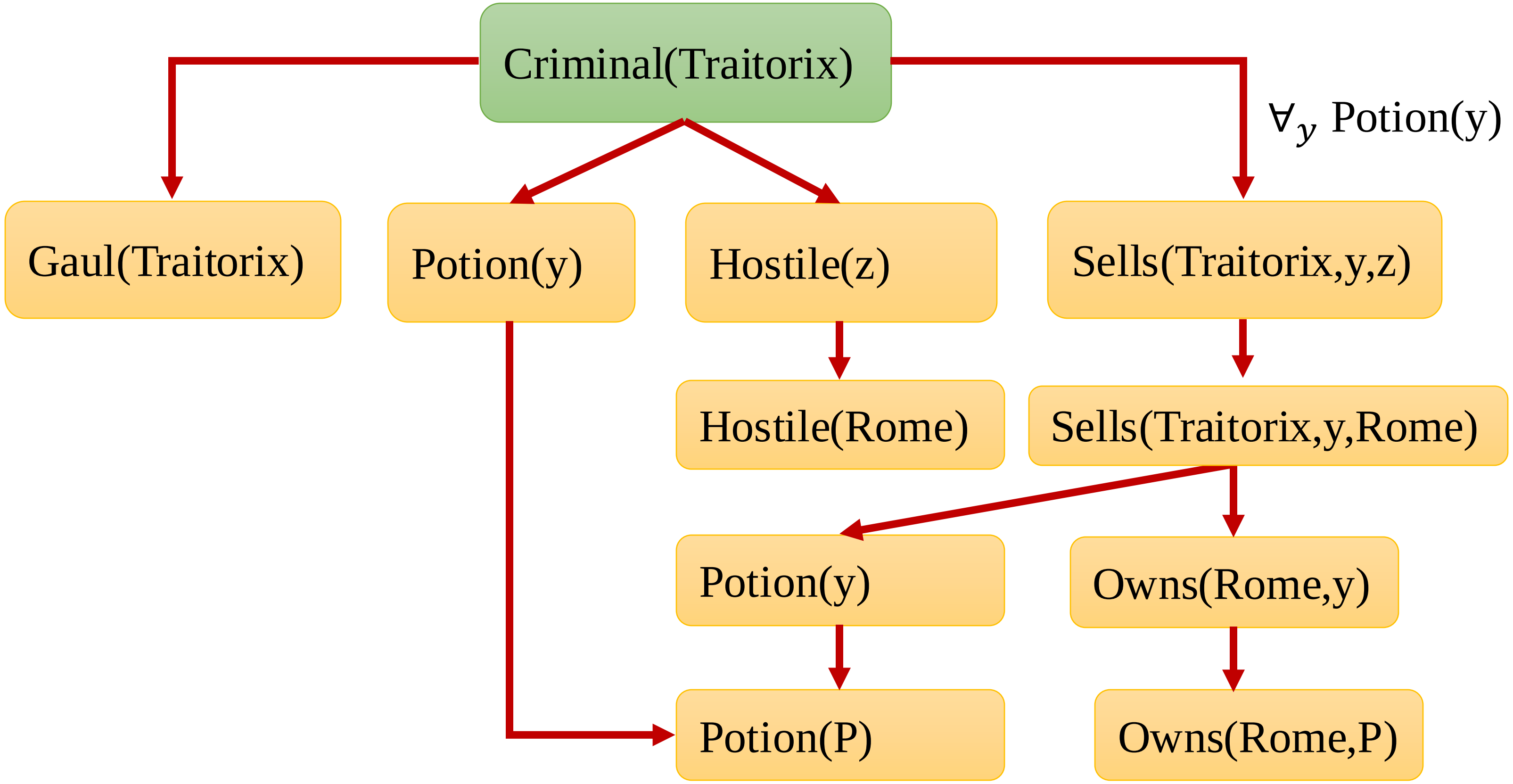


Reasoning in FOL: Backward Chaining

- $\forall_x \forall_y \forall_z$ Gaul(x) ∧ Potion(y) ∧ Sells(x,y,z) ∧ hostile(z) → Criminal(x)
- $\forall_y \forall_z$ Gaul(Traitorix) ∧ Potion(y) ∧ Sells(Traitorix,y,z) ∧ hostile(z) → Criminal(Traitorix)

Hostile(Rome)

$\forall_y$ Potion(y) ∧ Owns(Rome,y) → Sells(Traitorix, y, Rome)



Example2: Reasoning in FOL

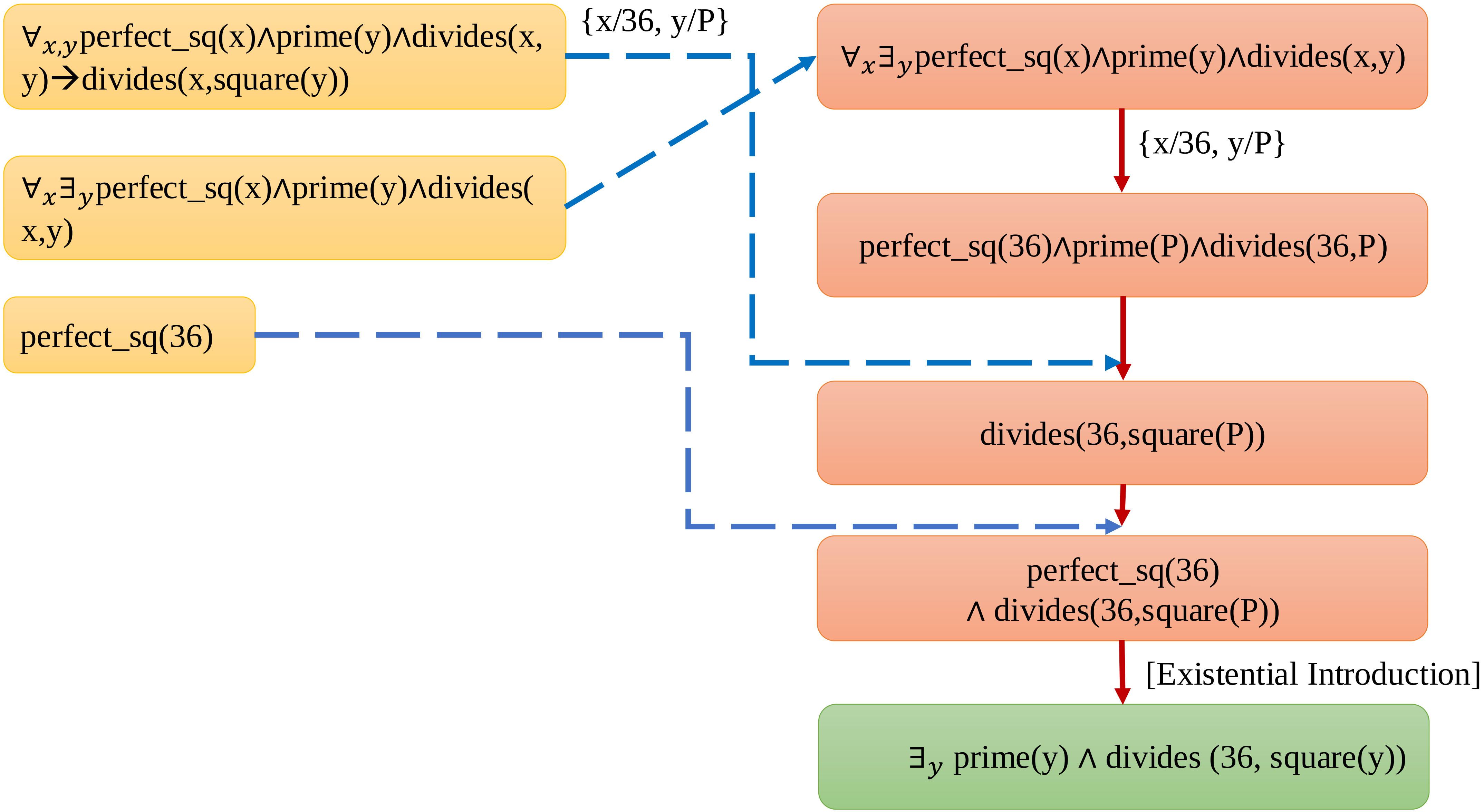
- If a perfect square is divisible by a prime, then it is also divisible by square of that prime.
- Every perfect square is divisible by some prime.
- 36 is a perfect square.

- Does there exist a prime such that square of that prime divides 36?

Example2: Representation in FOL

- If a perfect square is divisible by a prime, then it is also divisible by a square of that prime
  - $\forall_{x,y}$(perfect_sq(x)∧prime(y)∧divides(x,y)→divides(x,square(y)))
- Every perfect square is divisible by some prime
  - $\forall_{x}\exists_{y}$(perfect_sq(x)∧prime(y)∧divides(x,y))
- 36 is a perfect square
  - perfect_sq(36)
- Does there exist a prime such that the square of that prime divides 36?
  - $\exists_{y}$ (prime(y) ∧ divides (36, square(y)))

Reasoning in FOL: Forward Chaining



Thank You