Artificial Intelligence
Chapter 15.

The Predicate Calculus

15.1 Motivation
- Propositional calculus
  - Expressional limitation
  - Atoms have no internal structures.
- First-order predicate calculus
  - has names for objects as well as propositions.
  - Symbols
    - Object constants
    - Relation constants
    - Function constants
  - Other constructs
  - Refer to objects in the world
  - Refer to propositions about the world

15.2 The Language and its Syntax
- Components
  - Infinite set of object constants
    - Aa, 125, 23B, Q, John, EiffelTower
  - Infinite set of function constants
    - fatherOf, distanceBetween, times
  - Infinite set of relation constants
    - B17, Parent, Large, Clear, X11
  - Propositional connectives
    - ¬, ∧, ∨, ⊃, ☐
  - Delimiters
    - (, [, ], (separator)
15.2 The Language and its Syntax

- Terms
  - Object constant is a term
  - Functional expression
    - fatherOf(John, Bill), times(4, plus(3, 6)), Sam
- wffs
  - Atoms
    - Relation constant of arity n followed by n terms is an atom (atomic formula)
    - An atom is a wff.
    - Greaterthan(7,2), P(A, B, C, D), Q
  - Propositional wff
    - [Greaterthan n(7, 2) ∧ Lessthan(1, 5, 4)] ∨ ¬Brother(John, Sam) ∨ P

15.3 Semantics

- Worlds
  - Individuals
    - Objects
    - Concrete examples: Block A, Mt. Whitney, Julius Caesar, …
    - Abstract entities: 7, set of all integers, …
    - Fictional/invented entities: beauty, Santa Claus, a unicorn, honesty, …
  - Functions on individuals
    - Map n tuples of individuals into individuals
  - Relations over individuals
    - Property: relation of arity 1 (heavy, big, blue, …)
    - Specification of n-ary relation: list all the n tuples of individuals

15.3 Semantics

- Interpretations
  - Assignment: maps the followings
    - object constants into objects in the world
    - n-ary constants into n-ary functions
    - n-ary relation constants into n-ary relations
    - called denotations of corresponding predicate-calculus expressions
  - Domain
    - Set of objects to which object constant assignments are made

- True/False values

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>C</td>
<td>Floor</td>
</tr>
<tr>
<td>F1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>On</td>
<td>On={&lt;B,A&gt;, &lt;A,C&gt;, &lt;C, Floor&gt;}</td>
<td></td>
</tr>
<tr>
<td>Clear</td>
<td>Clear={&lt;B&gt;}</td>
<td></td>
</tr>
</tbody>
</table>

Table 15.1 A Mapping between Predicate Calculus and the World

Determination of the value of some predicate-calculus wffs

- On(A,B) is False because <A,B> is not in the relation On.
- Clear(B) is True because <B> is in the relation Clear.
- On(C,F1) is True because <C,Floor> is in the relation On.
- On(C,F1) ∧ ¬On(A,B) is True because both On(C,F1) and ¬On(A,B) are True.
15.3 Semantics

- Models and Related Notions
  - An interpretation satisfies a wff
  - wff has the value True under that interpretation
  - Model of wff
  - An interpretation that satisfies a wff
  - Valid wff
    - Any wff that has the value True under all interpretations
  - inconsistent/unsatisfiable wff
    - Any wff that does not have a model
  - Δ logically entails ω (Δ |= ω)
    - A wff ω has value True under all of those interpretations for which each of the wffs in a set Δ has value True
  - Equivalent wffs
    - Truth values are identical under all interpretations

(C) 2000 SNU CSE Artificial Intelligence Lab (SCAI) 9

15.4 Quantification

- Finite domain
  - Clear(B1) ∧ Clear(B2) ∧ Clear(B3) ∧ Clear(B4)
  - Clear(B1) ∨ Clear(B2) ∨ Clear(B3) ∨ Clear(B4)
- Infinite domain
  - Problems of long conjunctions or disjunctions → impractical
- New syntactic entities
  - Variable symbols
    - consist of strings beginning with lowercase letters
  - Term
  - Quantifier symbols → give expressive power to predicate-calculus
    - ∀: universal quantifier
    - ∃: existential quantifier

(C) 2000 SNU CSE Artificial Intelligence Lab (SCAI) 11

15.3 Semantics

- Knowledge
  - Predicate-calculus formulas
    - represent knowledge of an agent
  - Knowledge base of agent
    - Set of formulas
    - The agent knows ω = the agent believes ω

Floor

Figure 15.2 Three Blocks-World Situations
(C) 2000 SNU CSE Artificial Intelligence Lab (SCAI) 10

15.4 Quantification

- (∀ζ)ω, (∃ζ)ω : wff
  - ω: wff → within the scope of the quantifier
  - ζ: quantified variable
- Closed wff (closed sentence)
  - All variable symbols besides ζ in ω are quantified over in ω
    - (Ax)[P(x) ⊨ R(x)], (Ex)[P(x) ⊨ (Ey)[R(x, y) ⊨ S(f(x))]]
  - Property
    - (∀x)(∀y)ω = (∀y)(∀x)ω = (∀x, y)ω
    - (∃x)(∀y)ω ≠ (∃y)(∀x)ω
- First-order predicate calculi
  - restrict quantification over relation and function symbols

(C) 2000 SNU CSE Artificial Intelligence Lab (SCAI) 12
15.5 Semantics of Quantifiers

- Universal Quantifiers
  - $(\forall \zeta) \alpha(\zeta) = True$
    - $\alpha(\zeta)$ is True for all assignments of $\zeta$ to objects in the domain
  - Example: $(\forall x)[On(x, C) \supset \neg Clear(C)]$ in Figure 15.2
    - $x$: A, B, C, Floor
    - Investigate each of assignments in turn for each of the interpretations

- Existential Quantifiers
  - $(\exists \zeta) \alpha(\zeta) = True$
    - $\alpha(\zeta)$ is True for at least one assignment of $\zeta$ to objects in the domain

(C) 2000 SNU CSE Artificial Intelligence Lab (SCAI) 13

15.5 Semantics of Quantifiers

- Useful Equivalences
  - $\neg (\forall \zeta) \alpha(\zeta) \equiv (\exists \zeta) \neg \alpha(\zeta)$
  - $\neg (\exists \zeta) \alpha(\zeta) \equiv (\forall \zeta) \neg \alpha(\zeta)$
  - $(\forall \zeta) \alpha(\zeta) \equiv (\forall \eta) \alpha(\eta)$

- Rules of Inference
  - Propositional-calculus rules of inference $\rightarrow$ predicate calculus
    - Introduction and elimination of modus ponens $\wedge$
    - Introduction of $\lor$
    - $\neg$ elimination
    - Resolution
  - Two important rules
    - Universal instantiation (UI)
    - Existential generalization (EG)

(C) 2000 SNU CSE Artificial Intelligence Lab (SCAI) 14

15.6 Predicate Calculus as a Language for Representing Knowledge

- Conceptualizations
  - Predicate calculus
    - Language to express and reason the knowledge about real world
    - Represented knowledge: explored throughout logical deduction
  - Steps of representing knowledge about a world
    - To conceptualize a world in terms of its objects, functions, and relations
    - To invent predicate-calculus expressions with objects, functions, and relations
    - To write wffs satisfied by the world: wffs will be satisfied by other interpretations as well

(C) 2000 SNU CSE Artificial Intelligence Lab (SCAI) 15

15.6 Predicate Calculus as a Language for Representing Knowledge

- Conceptualizations
  - Predicate calculus
    - Language to express and reason the knowledge about real world
    - Represented knowledge: explored throughout logical deduction
  - Steps of representing knowledge about a world
    - To conceptualize a world in terms of its objects, functions, and relations
    - To invent predicate-calculus expressions with objects, functions, and relations
    - To write wffs satisfied by the world: wffs will be satisfied by other interpretations as well

(C) 2000 SNU CSE Artificial Intelligence Lab (SCAI) 16
15.6 Predicate Calculus as a Language for Representing Knowledge

- Usage of the predicate calculus to represent knowledge about the world in AI
  - John McCarthy (1958): first use
    - CYC project
    - represent millions of commonsense facts about the world
  - Nilsson 1991: discussion of the role of logic in AI
  - Genesereth & Nilsson 1987: a textbook treatment of AI based on logic

- Examples
  - Examples of the process of conceptualizing knowledge about a world
  - Agent: deliver packages in an office building
    - Package(x): the property of something being a package
    - Inroom(x, y): certain object is in a certain room
    - Relation constant Smaller(x, y): certain object is smaller than another certain object
    - “All of the packages in room 27 are smaller than any of the packages in room 28”
    \[(\forall x, y)\{\text{Package}(x) \land \text{Package}(y) \land \text{Inroom}(x,27) \land \text{Inroom}(y,28) \Rightarrow \text{Smaller}(x, y)\}\]

15.6 Predicate Calculus as a Language for Representing Knowledge

- “Every package in room 27 is smaller than one of the packages in room 29”
  \[(\exists y)(\forall x)\{\text{Package}(x) \land \text{Package}(y) \land \text{Inroom}(x,27) \land \text{Inroom}(y,28) \Rightarrow \text{Smaller}(x, y)\}\]
  \[(\forall x)(\exists y)\{\text{Package}(x) \land \text{Package}(y) \land \text{Inroom}(x,27) \land \text{Inroom}(y,28) \Rightarrow \text{Smaller}(x, y)\}\]

- Way of stating the arrival time of an object
  - Arrived(x,z)
  - X: arriving object
  - Z: time interval during which it arrived
  - “Package A arrived before Package B”
    \[(\exists z1, z2)\{\text{Arrived}(A,z1) \land \text{Arrived}(B,z2) \land \text{Before}(z1, z2)\}\]

- Temporal logic: method of dealing with time in computer science and AI

- Difficult problems in conceptualization
  - “The package in room 28 contains one quart of milk”
    - Mass nouns
    - Is milk an object having the property of being white?
    - What happens when we divide a quart into two pints?
    - Does it become two objects, or does it remain as one?

- Extensions to the predicate calculus
  - allow one agent to make statements about the knowledge of another agent
  - “Robot A knows that Package B is in room 28”
Additional Readings

- McDermott & Doyle 1980: discussion about
  - the use of logical sentences to represent knowledge
  - the use of logical inference procedures to do reasoning
- Tarski 1935, Tarski 1956: Tarskian semantics
  - Controversy about mismatch between the precise semantics of logical languages
- Agre & Chapman 1990
  - Indexical functional representations
- Enderton 1972, Pospesel 1976
  - Book on logic
- Barwise & Etchemendy 1993
  - Readable overview on logic