Artificial Intelligence
Chapter 13
The Propositional Calculus

Biointelligence Lab
School of Computer Sci. & Eng.
Seoul National University

Outline

- Using Constraints on Feature Values
- The Language
- Rules of Inference
- Definition of Proof
- Semantics
- Soundness and Completeness
- The PSAT Problem
- Other Important Topics

13.1 Using Constraints on Feature Values

- Description and Simulation
  - Description
    - Binary-valued features on what is true about the world and what is not true
    - easy to communicate
    - In cases where the values of some features cannot be sensed directly, their values can be inferred from the values of other features
  - Simulation
    - Iconic representation
    - more direct and more efficient

13.1 Using Constraints on Feature Values (Cont'd)

- Difficult or impossible environment to represent iconically
  - General laws, such as "all blue boxes are pushable"
  - Negative information, such as "block *A* is not on the floor" (without saying where block *A* is)
  - Uncertain information, such as "either block *A* is on block *B* or block *A* is on block *C*"
- Some of this difficult-to-represent information can be formulated as *constraints* on the values of features
  - These constraints can be used to infer the values of features that cannot be sensed directly.
- Reasoning
  - inferring information about an agent's personal state

13.1 Using Constraints on Feature Values (Cont'd)

- Applications involving reasoning
  - Reasoning can enhance the effectiveness of agents
  - To diagnose malfunction in various physical systems
    - represent the functioning of the systems by appropriate set of features
    - Constraints among features encode physical laws relevant to the organism or device.
    - features associated with "causes" can be inferred from features associated with "symptoms,"
    - *Expert Systems*

13.1 Using Constraints on Feature Values (Cont'd)

- Motivating Example
  - Consider a robot that is able to life a block, if that block is liftable and the robot's battery power source is adequate
  - If both are satisfied, then when the robot tries to life a block it is holding, its arm moves.
    - x_1 (*BAT_OK*)
    - x_2 (*LIFTABLE*)
    - x_3 (*MOVES*)
  - constraint in the language of the propositional calculus

    $$\text{BAT\_OK} \wedge \text{LIFTABLE} \supset \text{MOVES}$$

13.1 Using Constraints on Feature Values (Cont'd)

- *Logic* involves
  - A *language* (with a *syntax*)
  - *Inference rule*
  - *Semantics* for associating elements of the language with elements of some subject matter
- Two logical languages
  - *propositional calculus*
  - *first-order predicate calculus* (*FOPC*)

13.2 The Language

- Elements
  - *Atoms*
    - two distinguished atoms T and F and the countably infinite set of those strings of characters that begin with a capital letter, for example, P, Q, R, …, P1, P2, ON_A_B, and so on.
  - *Connectives*
    - $\vee$, $\wedge$, $\supset$, and $\neg$, called "or", "and", "implies", and "not", respectively.
  - *Syntax* of *well-formed formula* (*wff*), also called sentences
    - Any atom is a wff.
    - If w_1 and w_2 are wffs, so are $w_1 \vee w_2$, $w_1 \wedge w_2$, $w_1 \supset w_2$, $\neg w_1$.
    - There are no other wffs.

13.2 The Language (Cont'd)

- *Literal*
  - atoms and a ¬ sign in front of them
- *Antecedent* and *Consequent*
  - In $w_1 \supset w_2$, w_1 is called the antecedent of the implication.
  - w_2 is called the *consequent* of the implication.
- Extra-linguistic separators, (and)
  - group wffs into (sub) wffs according to the recursive definitions.

13.3 Rule of Inference

- Ways by which additional wffs can be produced from other ones
- Commonly used rules
  - *modus ponens*: wff w_2 can be inferred from the wffs w_1 and $w_1 \supset w_2$
  - $\wedge$ introduction: wff $w_1 \wedge w_2$ can be inferred from the two wffs w_1 and w_2
  - commutativity $\wedge$: wff $w_2 \wedge w_1$ can be inferred from the wff $w_1 \wedge w_2$
  - $\wedge$ elimination: wff w_1 can be inferred from the $w_1 \wedge w_2$
  - $\vee$ introduction: wff $w_1 \vee w_2$ can be inferred from either from the single wff w_1 or from the single wff w_2
  - $\neg$ elimination: wff w_1 can be inferred from the wff $\neg (\neg w_1)$.

13.4 Definitions of Proof

- Proof
  - The sequence of wffs $\{w_1, w_2, \ldots, w_n\}$ is called a *proof* of w_n from a set of wffs Δ iff each w_i is either in Δ or can be inferred from a wff earlier in the sequence by using one of the rules of inference.
- Theorem
  - If there is a proof of w_n from Δ, w_n is a theorem of the set Δ.
    - $\Delta \vdash w_n$
  - Denote the set of inference rules by the letter R.
    - w_n can be proved from Δ
    - $\Delta \vdash_R w_n$

Example

- Given a set, Δ, of wffs: {P, R, P ⊃ Q}, {P, P ⊃ Q, Q, R, Q ∧ R} is a proof of Q ∧ R.
- The concept of proof can be based on a partial order.

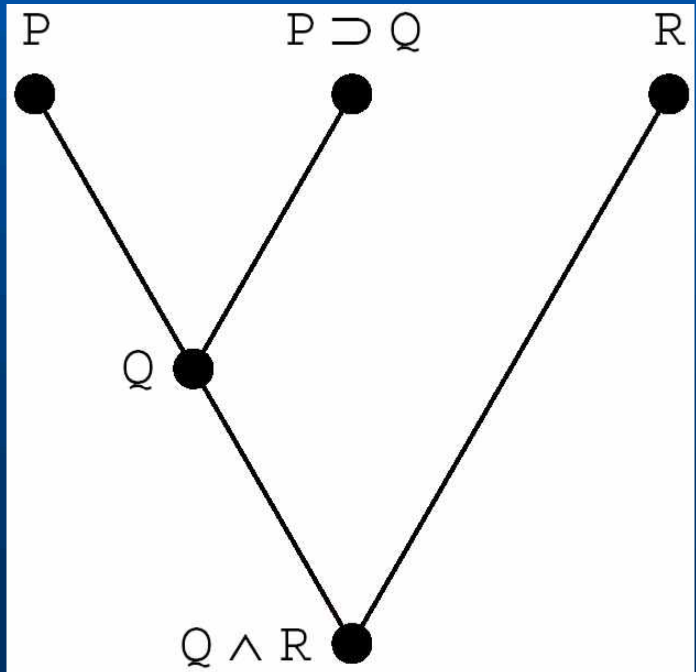

Figure 13.1 A Sample Proof Tree

13.5 Semantics

- Semantics
  - Has to do with associating elements of a logical language with elements of a domain of discourse.
  - Meaning
    - Such association
- Interpretation
  - An association of atoms with propositions
  - Denotation
    - In a given interpretation, the proposition associated with an atom

13.5 Semantics (Cont'd)

- Under a given interpretation, atoms have *values* – *True* or *False*.
- Special Atom
  - T : always has value True
  - F : always has value False
- An interpretation by assigning values directly to the atoms in a language can be specified – regardless of which proposition about the world each atom denotes.

Propositional Truth Table

- Given the values of atoms under some interpretation, use a truth table to compute a value for any wff under that same interpretation.
- Let w_1 and w_2 be wffs.
  - $(w_1 \wedge w_2)$ has *True* if both w_1 and w_2 have value *True*.
  - $(w_1 \vee w_2)$ has *True* if one or both w_1 or w_2 have value *True*.
  - $\neg w_1$ has value True if w1 has value *False*.
  - The semantics of $\supset$ is defined in terms of $\vee$ and $\neg$.

    Specifically, $(w_1 \supset w_2)$ is an alternative and equivalent form of $(\neg w_1 \vee w_2)$.

Propositional Truth Table (Cont'd)

- If an agent describes its world using *n* features and these features are represented in the agent's model of the world by a corresponding set of *n* atoms, then there are 2^n different ways its world can be.
- Given values for the *n* atoms, the agent can use the truth table to find the values of any wffs.
- Suppose the values of wffs in a set of wffs are given.
  - Do those values induce a unique interpretation?
  - Usually "No."
  - Instead, there may be many interpretations that give each wff in a set of wffs the value *True* .

Satisfiability

- An interpretation *satisfies* a wff if the wff is assigned the value True under that interpretation.
- *Model*
  - An interpretation that satisfies a wff
  - In general, the more wffs that describe the world, the fewer models.
- *Inconsistent* or *Unsatisfiable*
  - When *no* interpretation satisfies a wff, the wff is inconsistent or unsatisfiable.
  - e.g. F or $P \land \neg P$

Validity

- A wff is said to be *valid*
  - It has value *True* under *all* interpretations of its constituent atoms.
  - e.g.
    - $P \supset P$
    - T
    - $\neg (P \wedge \neg P)$
    - $Q \vee T$
    - $[(P \supset Q) \supset P] \supset P$
    - $P \supset (Q \supset P)$
  - Use of the truth table to determine the validity of a wff takes time exponential in the number of atoms

Equivalence

- Two wffs are said to be *equivalent* iff their truth values are identical under *all* interpretations.
- DeMorgan's laws

$$\neg(w_1 \vee w_2) \equiv \neg w_1 \wedge \neg w_2$$

$$\neg(w_1 \wedge w_2) \equiv \neg w_1 \vee \neg w_2$$

- Law of the contrapositive

$$(w_1 \supset w_2) \equiv (\neg w_2 \supset \neg w_1)$$

- If w_1 and w_2 are equivalent, then the following formula is valid:

$$(w_1 \supset w_2) \wedge (w_2 \supset w_1)$$

Entailment

- If a wff w has value True under all of interpretations for which each of the wffs in a set Δ has value True, Δ logically entails w and w logically follows from Δ and w is a logical consequence of Δ.
- e.g.
  - $\{P\} \models P$
  - $\{P, P \supset Q\} \models Q$
  - $F \models w$
  - $P \wedge Q \models P$

13.6 Soundness and Completeness

- If, for any set of wffs, Δ, and wff, w, $\Delta \vdash_R w$ implies $\Delta \models w$, the set of inference rules, *R*, is *sound*.
- If, for any set of wffs, Δ, and wff, *w*, it is the case that whenever $\Delta \models w$, there exist a proof of *w* from Δ using the set of inference rules, we say that *R* is *complete*.
- When inference rules are sound and complete, we can determine whether one wff follows from a set of wffs by searching for a proof.

13.6 Soundness and Completeness (Cont'd)

- When the inference rules are sound, if we can find a proof of *w* from Δ, *w* logically follows from Δ.
- When the inference rules are complete, we will eventually be able to confirm that *w* follows from Δ by using a complete search procedure to search for a proof.
- To determine whether or not a wff logically follows from a set of wffs or can be proved from a set of wffs is, in general, an NP-hard problem.

13.7 The PSAT Problem

- Propositional satisfiability (PSAT) problem: The problem of finding a model for a formula.
- *Clause*
  - A disjunction of literals
- *Conjunctive Normal Form (CNF)*
  - A formula written as a conjunction of clauses
- An exhaustive procedure for solving the CNF PSAT problem is to try systematically all of the ways to assign True and False to the atoms in the formula.
  - If there are n atoms in the formula, there are 2^n different assignments.

13.7 The PSAT Problem (Cont'd)

- Interesting Special Cases
  - 2SAT and 3SAT
  - *k*SAT problem
    - To find a model for a conjunction of clauses, the longest of which contains exactly *k* literals
  - 2SAT
    - Polynomial complexity
  - 3SAT
    - NP-complete
  - Many problems take only polynomial *expected* time.

13.7 The PSAT Problem (Cont'd)

- GSAT
  - Nonexhaustive, greedy, hill-climbing type of search procedure
  - Begin by selecting a random set of values for all of the atoms in the formula.
    - The number of clauses having value *True* under this interpretation is noted.
  - Next, go through the list of atoms and calculate, for each one, the increase in the number of clauses whose values would be True if the value of that atom were to be changed.
    - Change the value of that atom giving the largest increase
    - Terminated after some fixed number of changes
    - May terminate at a local maximum

13.8 Other Important Topics
13.8.1 Language Distinctions

- The propositional calculus is a formal language that an artificial agent uses to describe its world.
- Possibility of confusing the informal languages of mathematics and of English with the formal language of the propositional calculus itself.
  - $\vdash$ of $\{P, P \supset Q\} \vdash Q$
    - Not a symbol in the language of propositional calculus
    - A symbol in language used to talk about the propositional calculus

13.8.2 Metatheorems

- Theorems about the propositional calculus
- Important Theorems
  - Deductive theorem
    - If $\{w_1, w_2, \ldots, w_n\} \models w$, $(w_1 \wedge w_2 \wedge \ldots \wedge w_n) \supset w$ is valid.
  - Reductio ad absurdum
    - If the set Δ has a model but $\Delta \cup \{\neg w\}$ does not, then $\Delta \models w$.

13.8.3 Associative Laws and Distributive Laws

- Associative Laws

$$(w_1 \wedge w_2) \wedge w_3 \equiv w_1 \wedge (w_2 \wedge w_3)$$

$$(w_1 \vee w_2) \wedge w_3 \equiv w_1 \vee (w_2 \vee w_3)$$

- Distributive Laws

$$w_1 \wedge (w_2 \vee w_3) \equiv (w_1 \wedge w_2) \vee (w_1 \wedge w_3)$$

$$w_1 \vee (w_2 \wedge w_3) \equiv (w_1 \wedge w_2) \vee (w_1 \wedge w_3)$$