Chapter Overview

Rule-Based Reasoning with CLIPS

Introduction
Review Lessons "Rule-Based Reasoning", "CLIPS"

Forward Reasoning
Example: Flood Warning
Example: Boxer

Rete Algorithm
Example: Cats and Dogs

Backward Reasoning
Example: Dinner Choices
Example: Baseball

Chapter Review
Forward Reasoning

from data towards conclusions

**matching** known facts are compared with the premises of rules
if all premises of a rule are true, it is satisfied

**conflict resolution** if multiple rules are satisfied, one must be selected

**execution** the selected rule is applied
new facts or rules can be generated

**usage** need for quick response, few facts, few rules many acceptable conclusions

**domains** monitoring and real-time process control, synthesis; design, configuration, planning, scheduling
Flood Evacuation

example forward reasoning

problem
  flood warning and evacuation system for a flood-prone region

problem analysis
  • relatively small number of parameters and rules
  • relations known prior to execution
  • data acquisition automatically from sensors
  • continuous monitoring required

design decisions
  • forward reasoning
  • confidence factors
  • inference network
internal representation

- **rules**: rule number, parameters in the premises (upstream elements), parameters in the conclusions (downstream elements), premises, conclusions
- **parameters**: name, rules that modify the parameter, rules using the parameter, values and confidence factor (if known)
- **input data**: list of parameters used as inputs
- **conclusions**: output parameters
example forward reasoning

problem
system for arranging things to be packed into boxes

problem analysis

- relatively small number of parameters and rules
- not all relations known prior to execution
- synthesis problem: many possible solutions
- no optimal solution required
- pattern matching required for properties of objects
**design decisions**

- forward reasoning
- pattern matching system
- conflict resolution
  1) larger number of instantiated premises first
  2) ordering of input data
- grouping of rules, reflects overall packing strategy
- some simplifying assumptions
Pattern Matching

efficiency considerations

problem

pattern matching can be extremely inefficient:
all rules are compared with all the facts in
every cycle
rules * facts

observation

most of these comparisons are not necessary
in each step, only for changes in the fact base

solution

keep a list of satisfied rules
modify it only when changes occur due to
addition or deletion of facts
Rete Algorithm

efficient pattern matching

**basic idea**
keep track of satisfied rules
updates according to changes in the fact base

**pattern network**
set of trees from all premises of all rules

**join network**
connects leaf nodes of the trees that share
variables
checks consistency of variable bindings

**changed facts**
modify the corresponding entries in the
pattern and join network
Conflict Resolution

selecting the next rule

**number of rules** to be executed in one cycle
usually one, but several is possible (parallel execution)

**order of rules**
lowest number,
lowest number after current rule,
lowest numbered rule deriving a new fact, ...

**rule complexity** e.g. number of premises
most complex (specific) rule first, or most
generic (simplest) rule first

**order of facts**
lowest / highest number,
oldest / newest, ...

in practice combinations of the above criteria are used
Backward Reasoning

goal selection
take one goal and determine all rules capable of satisfying that goal
consequents of the rules must match the goal

matching
check applicable rules; if all premises are satisfied, a rule can be executed, and the goal is solved;
otherwise, new subgoals are created, or the user may provide inputs

usage
unknown response time, possibly many facts, many rules few acceptable conclusions

domains
classification, diagnosis;
inference networks for static knowledge,
otherwise pattern matching
Example: Dinner Choices

element backward reasoning

problem
select beverage and main course for a meal

problem analysis

- small number of parameters and rules
- relations known prior to execution
- inputs provided by the user upon request
- planning problem, few possible solutions
- no pattern matching necessary

design decisions

- backward reasoning
- inference network

If all inputs are provided in advance, forward reasoning can be applied
Example: Baseball

example backward reasoning with pattern matching

problem

choose a strategy in a particular situation here: signals from third base coach to batter and base runners

problem analysis

• larger number of parameters and variables
• few rules
• relations not fully known prior to execution
• inputs determined by the progress of the game
• decision problem, few satisfactory solutions
• pattern matching required
design decisions

- backward reasoning
- pattern matching

forward reasoning can be applied, too
potential problem: admissible, but inappropriate solution
Chapter Review

Rule-Based Reasoning with CLIPS

Introduction
Review Lessons ”Rule-Based Reasoning”, ”CLIPS”

Forward Reasoning
Example: Flood Warning Example: Boxer

Rete Algorithm
Example: Cats and Dogs

Backward Reasoning
Example: Dinner Choices Example: Baseball