Automatic Theorem Proving in Labelled Deductive Systems

Sonia Fabre
Department of Computer Science
University of Lund

Summary

- Involved Topics
- Goals
- Cases of study
- Design
- Implementation
- Tests
- Conclusions

Involved Topics

- Automatic Theorem Proving in Labelled Deductive Systems
  - Theorem Proving
  - Labelled Deductive Systems

Theorem Proving

- Theorem proving (or automated deduction)
  = logical deduction performed by machine
- At the intersection of several areas
  - Mathematics: original motivation and techniques
  - Logic: the framework and the reasoning techniques
  - Computer Science: the automation techniques
- Increase the certainty in some formal results

Labelled Deductive Systems (LDS)

- Basic idea: consider structured formulas
- Pairs <label : formula>
- Algebraic LDS
- Convenient use

LDS Proof System

- Three basic elements
  - Algebra of labels (A)
  - Logical Language (L)
  - Discipline of labelling (R)

Goals

- Implement a system capable of performing automatic theorem proving in LDS
- Study reasoning in active logics expressed as LDS
- Study two different systems: LS7 and Lmm
- Test the implementation to know more about the properties of the system

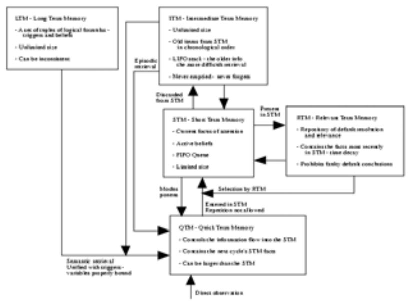
Cases of study (1)

- SL_7
  - Most complex version of Step Logic
  - Self-knowledge, time and retraction
  - Characterized by three elements: Language, Observations and Inference Rules
  - Special predicates: *K* , *Now*, and *Contra*

Cases of study (2)

- Memory model
  - Extension of SL_7 LDS to include all aspects of the memory model from [DMP86].
  - LTM, the long-term memory, which contains rules. Semantic retrieval is associative based on trigger formulae.
  - STM, the short term memory, which acts as the current focus of attention. All new inferences must include a formula from the STM.
  - QTM, the quick term memory, which is a technical device for buffering the next cycle's STM content.
  - RTM, the relevant term memory, which is the repository for default reasoning and relevance. It contains formulae, which have been pushed out of the STM but still may be important for default resolution.
  - ITM, the intermediate term memory, which contains all facts, which have been pushed out of the STM. The content of the ITM provides the history of the agents reasoning process. ITM provides support for goal directed behavior.

Cases of study (2)



Design (1)

- What do I have?
  - Formal definitions of the elements
  - Formal specifications of the inference rules
  - Some examples of proofs
- What do I need?
  - Elements: label, unit, formula and knowledge base
  - Functions: creation and management of the elements, implementation of the inference rules and the reasoning process

Design (2)

Labels manager	Tasks: creation and management of labels
Formulas manager	Tasks: creation and management of formulas
Unit manager	Tasks: creation and management of units
KB manager	Tasks: creation and management of the knowledge base
Inference Rules	Tasks: implementation of the inference rules
Reasoning Agent	Tasks: implementation of the process of reasoning

Elements: SL_7

- Unit
  - Label: a number that denotes the time point associated to the unit
  - Formula: a first order logic formula
- Knowledge base
  - Set of units

label : formula

3 : bird(Tweety)

label_1 : formula_1
label_2 : formula_2
label_3 : formula_3
...
label_n : formula_n

Functions: SL_7

- Inference rules
  The inference rules operate on the label part of a declarative unit as well as on the formula part.

$$\frac{i:\alpha,\ \mathrm{i}:\alpha \rightarrow \beta}{i+1:\beta}$$

- Reasoning Process
  The agent will try to apply the rules included in the inference rules set, generating a new state of the knowledge base with the obtained conclusions

i : ..., α
$i + 1$: ..., β

Elements: SL_{mm}

- Unit
  - Label:
    - Location
    - Trigger
    - Certainty
    - Time
    - Position
    - Time-left-in-rtm
  - Formula: a first order logic formula
- Knowledge base
  - Set of units

label : formula

(STM, _, C, i, i, 0) : bird(Tweety)

label_1 : formula_1
label_2 : formula_2
label_3 : formula_3
...
label_n : formula_n

Functions: L_{mm}

- Inference rules
  The same idea as in the other model

$$\frac{(STM,\varepsilon,c_1,i,p_1,R):\alpha,(STM,\varepsilon,c_2,i,p_2,R):\alpha \rightarrow \beta}{(QTM,\varepsilon,\min(c_1,c_2),i,0,0):\beta}$$

- Reasoning Process
  The same idea as in the other model

Implementation

- Two more attributes in unit
  - Number
  - Premisses
- Ansi Common Lisp
- Compatibilty packages
- Use of the system
  - GUI, not yet
  - The system is used from the prompt

Tests

- Every element in the system.
- Every inference rule in the system
- Reasoning agent: main procedure

Conclusions

- Modular system: Inference rules in the first LDS approach have been used in the second one. Only the way to manage the labels has been changed.
- Flexible system: It is possible to remove or add inference rules to the system, what allow us to study the behaviour of the system with different sets of rules.
- Ambigous definition of some concepts in the theory
- Some variations in the expected results of the execution of the system

Example: Execution

```
> (prover 'lmm set kb time)
[1 : (LTM, (BIRD $X), C, 0, 0, 0) : (FORALL ($X $I)  (=> (AND (BIRD $X) (NOW
    $I) (NOT (K (- $I 1) (NOT (FLIES TWEETY))))) (FLIES $X))) : (axiom)]
[2 : (LTM, (OSTRICH $X), C, 0, 0, 0) : (FORALL ($X) (=> (OSTRICH $X) (NOT
    (FLIES $X)))) : (axiom)]
[3 : (STM, 0, C, 0, 0, 20) : (NOW 0) : (AXIOM)]
[4 : (QTM, 0, C, 0, 0, 0) : (BIRD TWEETY) : (AXIOM)]
```

(prover Policy-label set-of-inference-rules Knowledge-base time)

Example: Different Results

```
[29 : (LTM, (BIRD $X), C, 4, 0, 0) : (FORALL ($X $I)  (=> (AND (BIRD $X) (NOW $I) (NOT
    (K (- $I 1) (NOT (FLIES TWEETY))))) (FLIES $X))) : (IL . 18)]
[30 : (LTM, (OSTRICH $X), C, 4, 0, 0) : (FORALL ($X) (=> (OSTRICH $X) (NOT (FLIES
    $X)))) : (IL . 19)]
[31 : (STM, 0, C, 4, 1, 20) : (BIRD TWEETY) : (IS . 20)]
[32 : (STM, 0, C, 4, 2, 20) : (FORALL ($I)  (=> (AND (BIRD TWEETY) (NOW $I) (NOT (K
    (- $I 1) (NOT (FLIES TWEETY))))) (FLIES TWEETY))) : (IS . 21)]
[33 : (STM, 0, C, 4, 4, 20) : (NOW 4) : (AXIOM)]
[34 : (STM, 0, C, 4, 4, 20) : (NOT (K (- 4 1) (NOT (FLIES TWEETY)))) : (IQS . 26)]
[35 : (STM, 0, C, 4, 4, 20) : (NOT (K (- 4 1) (FLIES TWEETY))) : (IQS . 27)]
[36 : (STM, 0, C, 4, 4, 20) : (FLIES TWEETY) : (IQS . 28)]
[37 : (QTM, 0, C, 4, 0, 0) : (OSTRICH TWEETY) : (AXIOM)]
[38 : (QTM, 0, C, 4, 0, 0) : (FORALL ($I)  (=> (AND (BIRD TWEETY) (NOW $I) (NOT (K (-
    $I 1) (NOT (FLIES TWEETY))))) (FLIES TWEETY))) : (SR 31 . 29)]
[39 : (QTM, 0, C, 4, 0, 0) : (FORALL ($I)  (=> (AND (BIRD TWEETY) (NOW $I) (NOT (K (-
    $I 1) (NOT (FLIES TWEETY))))) (FLIES TWEETY))) : (SR 36 . 29)]
[40 : (QTM, 0, C, 4, 0, 0) : (NOT (K 4 (NOT (FLIES TWEETY)))) : (NI . 32)]
[41 : (QTM, 0, C, 4, 0, 0) : (FLIES TWEETY) : (EMP 32 31 33 34)]
```


Example: Comparing with theory

```
[31 : (STM, 0, C, 4, 1, 20) : (BIRD TWEETY) : (IS . 20)]
...
[36 : (STM, 0, C, 4, 4, 20) : (FLIES TWEETY) : (IQS . 28)]
...
[38 : (QTM, 0, C, 4, 0, 0) : (FORALL ($I)  (=> (AND (BIRD TWEETY) (NOW $I) (NOT (K (-
    $I 1) (NOT (FLIES TWEETY))))) (FLIES TWEETY))) : (SR 31 . 29)]
...
[39 : (QTM, 0, C, 4, 0, 0) : (FORALL ($I)  (=> (AND (BIRD TWEETY) (NOW $I) (NOT (K (-
    $I 1) (NOT (FLIES TWEETY))))) (FLIES TWEETY))) : (SR 36 . 29)]
```

Semantic Retrieval is not applicable between units 36 and 29 in the theory.

Questions

?