ICOT Technical Memorandum: TM-0158

TM-0158

Toward Mechanization of Deductive, Inductive and Abductive Inference

by
K. Tsurumaki (NTT)
S. Kunifuji, H. Seki, T. Takewaki
and K. Furukawa (ICOT)

March, 1986

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato ku Tokyo 108 Japan

(03) 456-3191~5
Telex ICOT J32964

Institute for New Generation Computer Technology

Toward Mechanization of Deductive, Inductive and Abductive Inference

Susumu KUNIFUJI, Hirohisa SEKI,
Toshiaki TAKEWAKI, Koichi FURUKAWA
(ICOT)
Koji TSURUMAKI
(NTT)

[Abstract]

There exist three kinds of inference processes in the human problem-solving. They are deduction, induction and abduction [Kunifuji 85-2]. The research goal of the problem-solving and inference group of ICOT is to build a problem-solving and inference support system with such inference capabilities.

In the initial stage of the project, a knowledge acquisition support system was investigated in ICOT as a basic function of the knowledge base management system in the fifth generation systems. The system is implemented in DEC-10 Prolog on DEC2060. To support deductive and inductive inference-based knowledge acquisition, the system employs two kinds of knowledge acquisition functions, i.e. knowledge assimilation and knowledge accommodation (cf. Fig. 1). Knowledge assimilation [Bowen 82, Miyachi 84, Kunifuji 85-1] means the activity of adding new facts or rules to the knowledge base, without violating its consistency. Knowledge accommodation [Kitakami 84-1, Kunifuji 85-1] means consistently modifying the knowledge base, using model inference system [Shapiro 82], when adding new correct facts or rules to it. In the knowledge accommodation process, given added facts or rules are supposed to be absolutely true, while, in the knowledge assimilation process, given knowledge base is supposed to be consistent. The essential problem in assimilation and accommodation is to keep the entire knowledge base consistent and hopefully non-redundant. The system can manage several types of knowledge acquisition functions, shown in Fig. 2, Fig. 3, and Fig. 4 [Kitakami 84-2]. Note that all functions are implemented by using meta-programming techniques in Prolog.

In the intermediate stage of the project, the most attractive theme of the problem-solving and inference group is to mechanize abductive inference. As a first step toward mechanizing abduction, the authors are now building hypothesis generation and selection system [Tsurumaki 85], which is similar to Theorist [Poole 85]. The experimental system implemented in DEC-10 Prolog generates possible hypotheses from a set of tentative hypothetical formulas and select one to form consistent explanations of given observations. The differences between the authors' system and Theorist are as follows: (1) In our system, hypothses are represented in the is-a hierarchical structure and for given observations, appropriate hypotheses are derived based on that structure. On the other hand, in Theorist, hypotheses are not supposed to have such structure. (2) If multiple possible hypotheses are to be given by the system, our system asks new observations to the user in order to select one of such possible hypotheses. The most important thing is that meta-programming methodology enables the user easily to define the criteria of hypothesis generation and selection mechanism, shown in Fig. 5. The above-mentioned characteristics mean that the

system is "a workbench for thinking !" of hypothesis generation and selection. The system, demonstrated in Fig. 6, can solve an unknown ion identification problem.

The subjects for a future study are as follows: (a) establishing a logical approach to the problem of hypothesis selection, such as an algorithm for finding a query which discriminates competing hypotheses [Seki 85] ; (b) integrating an analogical reasoning function [Haraguchi 85] to the system, which enlarges reasoning capability in the system.

Reference

[Bowen 82] Bowen,K. A., Kowalski, R. A. : Amalgamating Language and Meta-language, in Logic Programming, Clark,K. L., Taernlund, S. -A. (Eds.), Academic Press, pp.153-172,1982.

[Haraguchi 85] Haraguchi, M., Arikawa, S.: Analogical Reasoning based on the Theory of Analogy, Res. Rep. R.I.F.I.S., Kyushu Univ., 1985.

[Kitakami 84-1] Kitakami, H., Kunifuji, S., Miyachi, T., Furukawa, K.: A Methodology for Implementation of a Knowledge Acquisition System, Proc. of IEEE 1984 International Symposium on Logic Programming, Atlantic City, U.S.A., pp.131-142, Feb. 1984.

[Kitakami 84-2] Kitakami, H., Kunifuji, S., Miyachi, T., Furukawa, K., Takeuchi, A., Miyazaki, T., Kondou, H., Ishii, S., Takewaki, T., and Ohki, M.: Demonstration of the KAISER System at the ICOT Open House in FGCS'84, ICOT TM-0081, Nov. 21,1984.

[Kunifuji 85-1] Kunifuji, S., Kitakami, H., Miyachi, T., Furukawa, K.: Knowledge Base Management in Logic Programming Language Prolog, Proc. of the Logic Programming Conference '85, Tokyo, July 1985.

[Kunifuji 85-2] Kunifuji, S.: Toward Mechanization of Deductive, Inductive and Abductive Inference, Journal of Japan Creativity Society No. 3, Part 2 The Road to Creative Science, Kyoritsu-shuppan, Oct. 1985.

[Miyachi 84] Miyachi, T., Kunifuji, S., Kitakami, H., Furukawa, K., Takeuchi, A., and Yokota, H.: A Knowledge Assimilation Method for Logic Databases, Proc. of IEEE 1984 International Symposium on Logic Programming, Atlantic City, U.S.A., Feb. 1984.

[Poole 85] Poole, D., Aleliunas, R., Goebel, R.: Theorist: a logical reasoning system for defaults and diagnosis, submitted as a chapter in the volume Knowledge Representation, Cercone, N. J. & McCalla,G.(eds.), IEEE Press, in preparation.

[Seki 85] Seki, H., Takeuchi, A.: An Algorithm for Finding a Query which Discriminates Competing Hypotheses, to appear in ICOT TR, 1985.

[Shapiro 82] Shapiro, E. Y.: Algorithmic Program Debugging, ACM Distinguished Dissertations, The MIT Press, 1982.

[Tsurumaki 85] Tsurumaki, K., Kunifuji, S., Furukawa, K.: Hypothesis Generating System Using Meta-programming Techniques, Proc. of Second Conference Japan Society of Software Science and Technology, Tokyo, Dec. 1985.

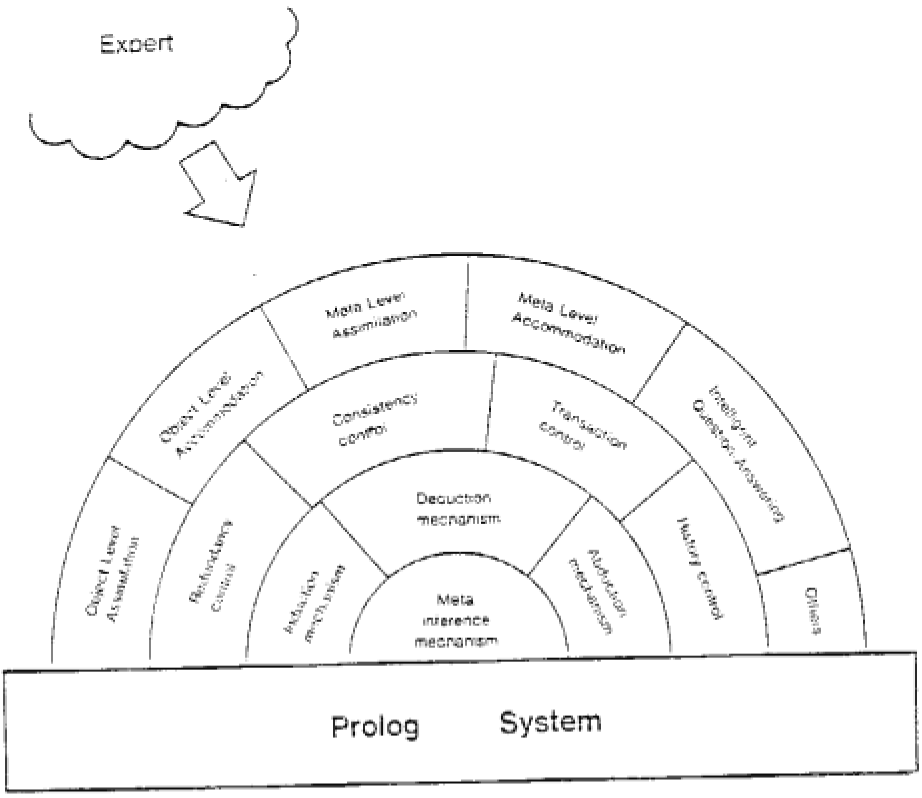

Fig. 1 Knowledge Acquisition Functions

Knowledge Base

(1) Constraint

is_inconsistent (insert, _):− s (X, [ ]), exist_variable (X).

(Any sentence generated by grammer rules must not have unistantiated variables.)

(2) Dictionary

```
n ([hermia |X], X).                     — noun
n ([forest |X], X).                     — noun
vi ([walks |X], X).                     — intransitive verb
vt ([loves |X], X).                     — transitive verb
p ([in |X], X).                         — preposition
np (X, Y) :— n (X, Y).                  — noun phrase
vp (X, Y) :— vi (X, Y).                 — verb phrase
vp (X, Y) :— vt (X, Z), n (Z, Y).       — verb phrase
pp (X, Y) :— p(X, Z), n(Z, Y).          — prepositional phrase
```

Fig. 2 Given Knowledge Base for Object-level Acquisition

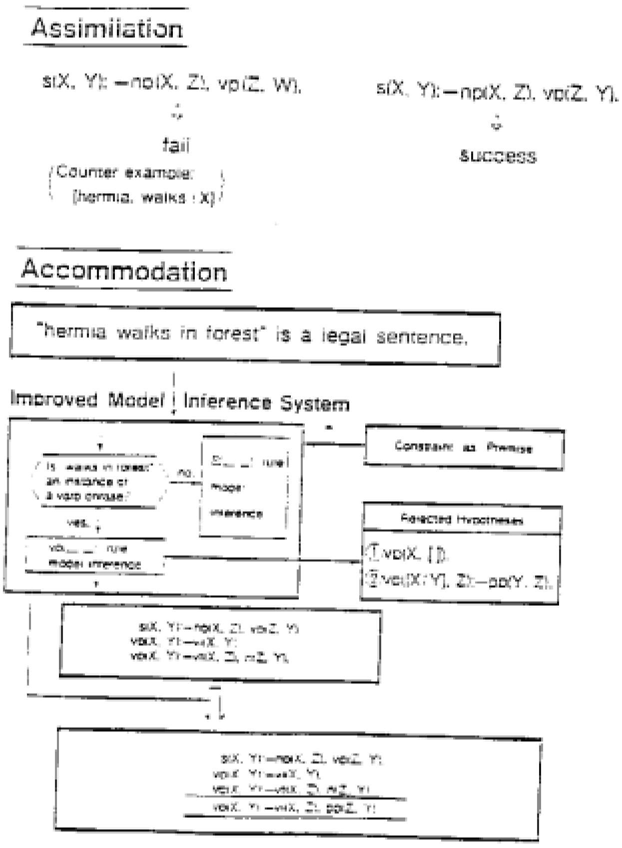

Fig. 3 Object-level Knowledge Assimilation and Accommodation

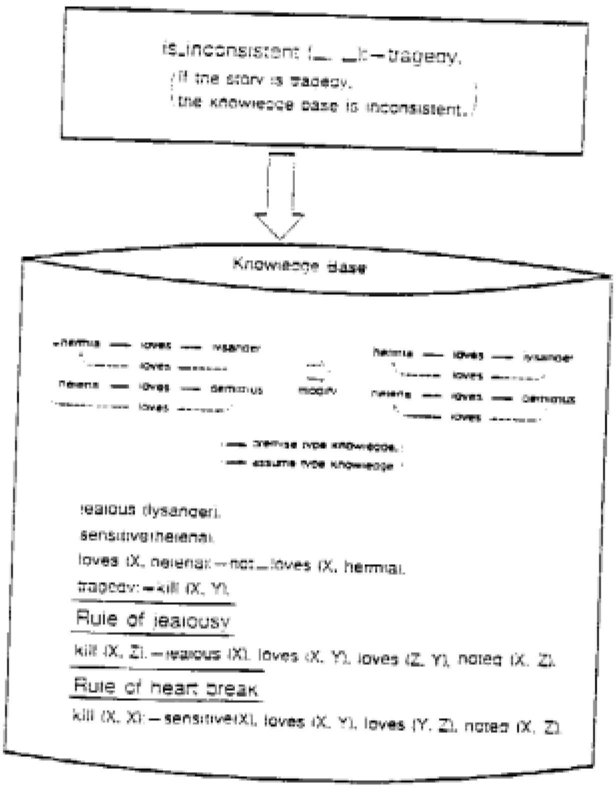

Fig. 4 Meta-level Knowledge Assimilation

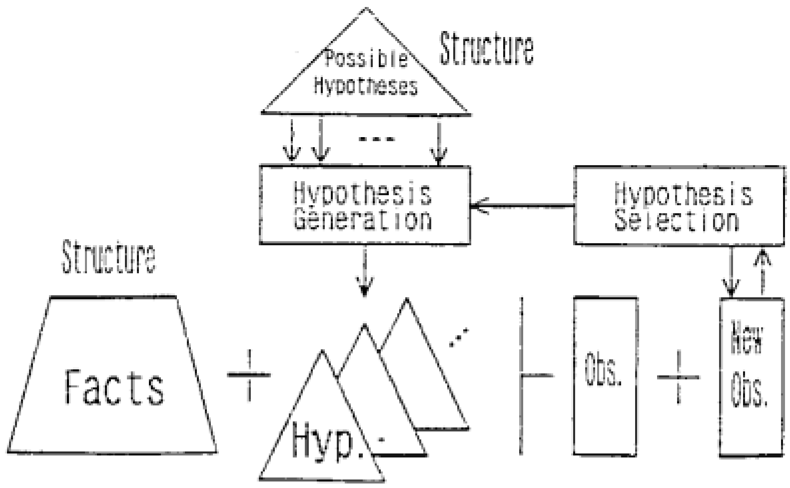

Fig. 5 Framework of Hypothesis Generation and Selection

```
| ?- start.

observations: precip_OH_ion(x),
              precip_S_ion(x),
              precip_SO4_ion(x).

Tentative Possible Explanations for the observations

     Pb

Is "precip_halogen_ion(x)" right (y/n) ?
>> n

Tentative Possible Explanations for the observations

     Ca & transition_element ;
     transition_element & alkaline_earths

Is "flame_color_orange(x)" right (y/n) ?
>> y

Is "precip_PO4_ion(x)" right (y/n) ?
>> y

Is "precip_CO3_ion(x)" right (y/n) ?
>> y

verified_explanation :
     Ca & transition_element

yes
| ?-
```

Fig. 6 Example of Ion Identification Problem