

## 4M-12

## 論証支援システムのための論理式エディタ

佐藤かおる 土屋恭子 小野越夫 沢村一 南俊朗  
(富士通株式会社)

## 1. はじめに

計算機科学や人工知能の分野において、計算機による問題解決やその支援の研究が盛んである。これらを実現する上で、「問題解決とは、把握した問題を適当な公理系の下で形式化し、論証を行うことである」と見做す論理学的方法の有効性が指摘されている(1)。しかし、このような過程の全てを自動的に行うことは非常に難しいため、まず問題の形式化や、論証に必要な論理式の操作を支援するような論証支援システムが望まれる。

このような論証支援システムは、既に研究されているが(2他)、いずれも特定の論理に対して支援を行うものである。一方、問題を形式化するための論理は問題に応じて多種多様に存在している。これら多種多様な論理の下での論証支援には、それぞれの論理に対して支援システムを構築するのではなく、汎用の論証支援システムを提供することが有効であると考えられる。

また、論証の支援には、問題の形式化、すなわち証明を行うのに必要な論理式の入力、編集の支援と、入力論理式に対する証明の支援の2つが考えられる。

本論文では、汎用の論証支援システムにおいて前者の支援を会話的に行うことを目的とする論理式エディタ(3)について報告する。後者の支援に関しては、(4)を参照されたい。

なお、本研究は、新世代コンピュータの開発の一環として、ICOTの委託によって行ったものである。

## 2. 論理式エディタ

汎用の論証支援システムにおける論理式エディタとしては、任意の論理系の下での論理式操作を可能とするために、論理式の一般的構造を対象とした構造操作機能と、論理系を規定する機能とが必要である。

一般的に、論理式は再帰的な木構造を持っているので、その構造操作は木構造操作と見做すことができる。更に、論理式特有の記号操作も行えることが望ましい。また、論理系を規定するには、論理式の構文の定義と書き換え規則の定義が必要である。

我々の開発している論理式エディタは、これらの要件を満たすために、論理式の構造、論理式の構文定義、書き換え規則の各々に対する操作機能を提供する。なお、現在のところ、論理式の構文定義機能としては、記号定義機能だけを提供している。

図1に、論理式エディタの構成を示す。論理式とその論理式に対して使用される記号定義及び書き換え規則は、論理式の定義内に、記号定義、書き換え規則の定義を指定することによって関係づけられる。

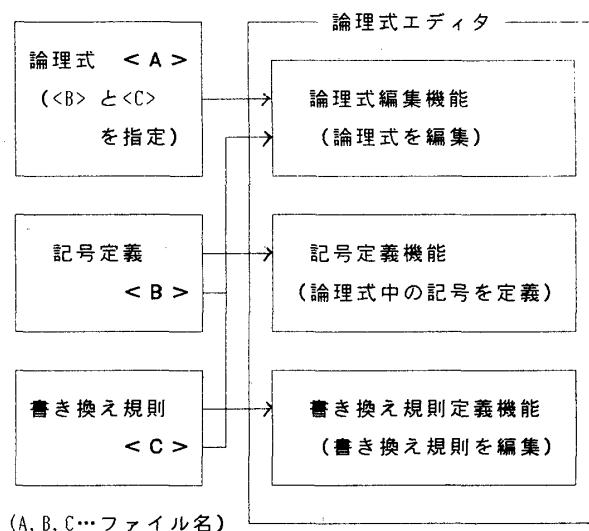


図1. 論理式エディタの構成図

以下、論理式編集機能を中心に述べる。

## 3. 論理式編集機能

論理式編集機能は、木構造の操作を行うのに必要な機能と、論理式を操作する上で特に必要な機能を有する。

## 3.1 木構造操作機能

編集対象である論理式の構文は、現在のところ、PROLOGの項である。木構造操作を行うために、この項を木構造を持つ内部表現に変換して操作を行う。

ウィンドウには、図2に示すように、操作対象の論理式

の木構造表現が表示され、ユーザは木構造を認識しながら編集を行うようになっている。ノード単位に、削除、挿入、置換、複写、移動、項の探索等の操作が可能である。

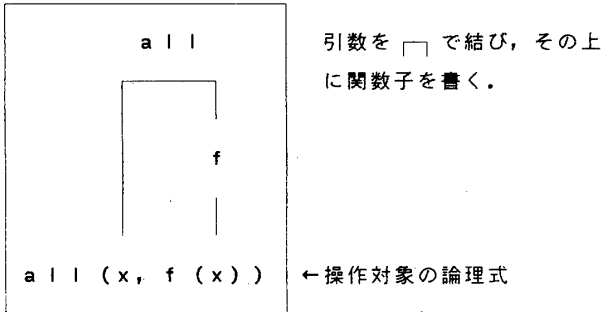


図 2. 論理式の木構造表現  
( a l l ( x , f ( x ) ) の木構造表現)

### 3.2 論理式操作機能

論理式エディタは、上述した木構造エディタとしての機能に加え、論理式を操作するという特定の目的のための機能も有する。このような機能としては、変数記号に対する任意の項の代入、書き換え規則を用いた論理式の同値変形等がある。これらの機能を実現するために必要な基本機能は、記号の認識と、論理式と書き換え規則とのパターン・マッチの機能である。

#### (1) 記号の認識

記号の属性は、構文的型と型によって表されている。構文的型としては、変数、定数、関数子、論理演算子、演算子がある。型とは、ブール、自然数等、その記号の持つ値の属性である。これらの記号情報は、ユーザ自身によって記号定義機能を用いて定義される。論理式エディタは、この記号定義情報に基づいて、論理式中の記号の属性を認識する。

#### (2) 書き換え規則とのパターン・マッチ

書き換え規則は、論理式の同値変形を行う際に必要な情報である。書き換え規則は、“ A = B ( A , B は、論理式のパターン ) ” の形式で作成する。この書き換え規則は、A のパターンの論理式を B のパターンの論理式に変換するということを意味する。

論理式エディタにおける論理式の同値変形は、指定された論理式と書き換え規則の左辺のパターンとを比較し、パターン・マッチする規則が存在したならば、その規則の右辺のパターンと置き換えることによって実現される。

#### (3) 論理式操作の例

論理式操作の例を図 3 に示す。この例では、“ P → Q

( P ならば Q である ) ” という論理式が、同値変形を用いて “ ~ Q → ~ P ( Q でないならば、 P でない ) ” に変換されている。論理式 “ P → Q ” に対して、記号 “ x ” , “ y ” が変数として定義され、規則 R1 と R2 が書き換え規則として定義されているものとする。

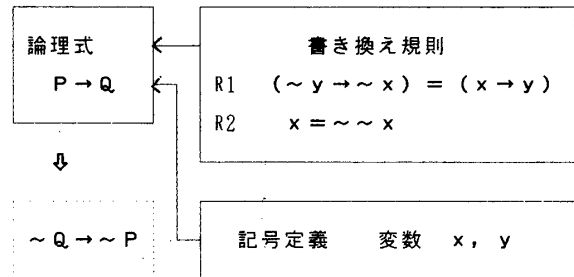


図 3. 対偶の法則を導く例

#### 操作手順

- (1) 論理式の → の左辺を同値変形する。  

$$P \rightarrow Q \Leftrightarrow \sim \sim P \rightarrow Q$$
 ( R2 にパターン・マッチ )
- (2) 論理式の → の右辺を同値変形する。  

$$\sim \sim P \rightarrow Q \Leftrightarrow \sim \sim P \rightarrow \sim \sim Q$$
 ( R2 にパターン・マッチ )
- (3) 論理式全体を同値変形する。  

$$\sim \sim P \rightarrow \sim \sim Q \Leftrightarrow \sim Q \rightarrow \sim P$$
 ( R1 にパターン・マッチ )

#### 4. おわりに

現在、我々は、汎用論証支援システムの構築を目的とし、論証に必要な論理式の入力、編集を支援する論理式エディタを開発している。本論文では、論理式の構造操作及び同値変形等の操作を行うための論理式編集機能を中心に報告した。今後の課題として、論理式の構文定義を可能とすること、使用経験を通して機能拡張を行うこと等が挙げられる。

#### 参考文献

- (1) R.Turner:Logics for artificial intelligence, Ellis Horwood Limited, 1984.
- (2) J.Ketonen and J.S.Weening: EKL-An interactive proof checker, user's reference manual, Dept.of Computer Science, Stanford university, 1984.
- (3) 沢村一他:論理式エディタ, 構造エディタに関するワークショップ ( W S E ), 日本ソフトウェア科学会, 1986.
- (4) 南俊朗他:論証支援のための証明コンストラクタ, 情報処理学会第 33 回全国大会, 1986.