構造帰納法の証明の分析

2018SE107 吉見颯
指導教員：横山哲郎

1　はじめに

プログラムの正確性を保証するためには,「証明」を行う必要がある．プログラムのテストでは，テストしなければならない状態が多く，テストケースが膨大な数になる可能性があるが，証明を行えば，1 回でプログラムの正確性を保証可能だからである．証明を行うことは，複雑なプログラムや信頼性の高さが問われるプログラムにおいて，より必要なものになる．本研究では，様々なプログラムの性質を帰納法で証明し，帰納法について理解を深める．チューリング機械の計算の正確性やラムダ式の型判定に関する性質を対象とする．研究課題としては，プログラムの性質を証明する上で，数学的帰納法や累積帰納法，構造帰納法，導出に関する帰納法などの帰納法を間接的に用いることが可能だが，それぞれのプログラムの性質の証明において直接的に用いることができる帰納法はどれなのかを確かめる．また，様々な帰納法が整礎帰納法の特殊ケースになっていることの確認を行う．

2　導出システム

導出システムとは，論理式，プログラム，型などの議論の対象に対する様々な判断を推論規則に従って導くためのシステムである．本研究では，自然数の加算，乗算の概念を導出システム（以後，導出システム Nat と呼ぶ）として与える．

2.1　自然数の集合

導出システムにおいて自然数は，`S(`$\cdots$ `S(Z)`$\cdots$`)` と表現する．すなわち，全ての自然数を `S` と `Z` の 2 つの記号のみで表現する．以下に例を示す．

$$0, 1, \ldots, 5, \ldots \Rightarrow \texttt{Z}, \texttt{S(Z)}, \ldots, \texttt{S(S(S(S(S(Z)))))}, \ldots$$

この記法により，加算，乗算の概念を少ない推論規則で表すことが可能になる．この記法で表現された自然数を,「ペアノ自然数」と呼ぶ.

本研究では，BNF を用いてペアノ自然数全体の集合 Nat の構文を定義する．複雑な構造を持つプログラムには，より厳密な指定ができる記法が必要なためである．BNF(バッカス–ナウア記法) とは，プログラミング言語の構文定義をするときの標準的な記法である．以下に，BNF を用いたペアノ自然数全体の集合 Nat の構文を定義する．

定義 2.1　ペアノ自然数の集合 Nat を以下の構文で定義する．

$$n \in N \ ::= \ \texttt{Z} \mid \texttt{S}(n)$$

BNF では，”::=”は「〜は以下のもので構成される」，”｜”は「または」という意味である．これらを踏まえると以下のように解釈できる．

- `Z` は Nat の要素，つまりペアノ自然数である．
- `S` が Nat の要素 n を引数にとったものも Nat の要素である．

以降ではペアノ自然数の変数として n を用いる．さらに，添え字を用いて n_1, n_2, n_3 なども用いる．

2.2　判断の形式

ここでいう判断とは，具体的なふたつの自然数に対する加算または乗算とその結果を述べた文のことである．

定義 2.2　導出システム `Nat` では，以下の 2 つの判断のみ扱う．

- n_1 `plus` n_2 `is` n_3
- n_1 `times` n_2 `is` n_3

意味としては，それぞれ「自然数 n_1 と n_2 の和は n_3 である.」,「自然数 n_1 と n_2 の積は n_3 である.」となる．

2.3　推論規則と導出の記法

導出システムを完成させるには，各判断を導くための推論規則が必要となる．

「j_1ならば j_2」となる推論規則の j_1 を「前提」，j_2 を「結論」と呼ぶ．推論規則では，前提の判断 j_1 が既に導かれている場合のみ，j_2 の判断を導くことが可能である．以下のような推論規則があるとする．

[推論規則 X]J_1かつ J_2かつ$\cdots$かつ J_nならば J_0である．

このような推論規則を以降では次のように表現する．

$$\frac{J_1 \ J_2 \cdots J_n}{J_0}(X)$$

前提の数 n は，規則 P-ZERO のような前提のない規則の場合は 0 となる．

導出は導出木という記法で表現する．導出木は，判断をノード，結論となる判断を根とする木構造がとられる．一般的に，判断 J_i を具体化したもの J_i' を結論とする $D_i(i = 1, \ldots, n)$ が得られており，また，推論規則 *Foo* 中のパラメータを具体化したもの

$$\frac{J_1' \cdots J_n'}{J_0'}$$

が得られたとすると，

$$\frac{D_1 \cdots D_n}{J_0'} \mathit{Foo}$$

は J_0' を結論とする導出である．また，前提のない推論規則 Bar を具体化したもの

$$\frac{}{J_0'}Bar$$

は導出である．

以下に導出システム Nat の推論規則を与える．

$$\frac{}{\texttt{Z plus } n \texttt{ is } n}\text{(P-ZERO)}$$

$$\frac{n_1 \texttt{ plus } n_2 \texttt{ is } n}{\texttt{S(}n_1\texttt{) plus } n_2 \texttt{ is S(}n\texttt{)}}\text{(P-SUCC)}$$

$$\frac{}{\texttt{Z times } n \texttt{ is Z}}\text{(T-ZERO)}$$

$$\frac{n_1 \texttt{ times } n_2 \texttt{ is } n_3 \qquad n_2 \texttt{ plus } n_3 \texttt{ is } n_4}{\texttt{S(}n_1\texttt{) times } n_2 \texttt{ is } n_4}\text{(T-SUCC)}$$

3　メタ定理

メタ定理とは，導出システムについて数学的に証明された定理のことである．以下では，メタ定理の有用な証明法である「帰納法」を用いて，いくつかのメタ定理を証明していく．帰納法とは一般的に「集合 X の任意の元 x に対して ... である」という形式の命題を証明するための技法である．

4　帰納法を用いた証明

本稿では，数学的帰納法，累積帰納法，構造帰納法などの帰納法を例とともに示し，プログラムの性質を証明していく．この節では，構造帰納法を用いてペアノ自然数同士の加算について証明する．

4.1　構造帰納法

構造帰納法とは，ペアノ自然数や算術式などの BNF で定義される対象の性質を証明するのに用いることができる帰納法である．

4.2　ペアノ自然数に関する帰納法の原理

以下に，ペアノ自然数に関する帰納法の原理を示す．
命題 $P(n)$ について以下の 2 つの条件が成り立つとき，命題 $P(n)$ は全てのペアノ自然数 n に成り立つとする．

(i) $P(\texttt{Z})$ が成り立つ．

(ii) 任意のペアノ自然数 n について，$P(n)$ が成り立つならば，$P(\texttt{S(}n\texttt{)})$ も成り立つ．

4.3　具体例と考察

例として以下のペアノ自然数同士の加算に関する補題を構造帰納法を用いて証明する．

また，補題 1 のそれぞれの性質の証明法について比較する．

補題 1

1. 任意のペアノ自然数 n に対し，$\texttt{Z plus } n \texttt{ is } n$ である．
2. 任意のペアノ自然数 n に対し，$n \texttt{ plus Z is } n$ である．

証明

1. 推論規則 P-ZERO により，ペアノ自然数 n に関わらず，

$$\frac{}{\texttt{Z plus } n \texttt{ is } n}\text{P-ZERO}$$

という導出木が作れる．

2. ペアノ自然数に関する帰納法によって証明する．
   (i) $n \equiv \texttt{Z}$ の場合．つまり $\texttt{Z plus Z is Z}$ は，推論規則 P-ZERO によって導出できる．
   (ii) $n \equiv k$ の場合．つまり $k \texttt{ plus Z is } k$ が導出できると仮定する．$k \texttt{ plus Z is } k$ の導出を D とすると，

$$D = \frac{\vdots}{k \texttt{ plus Z is } k}$$

となる．このとき，推論規則 P-SUCC を用いると，

$$\frac{D}{\texttt{S(}k\texttt{) plus Z is S(}k\texttt{)}}\text{ P-SUCC}$$

となり，$\texttt{S(}k\texttt{) plus Z is S(}k\texttt{)}$ が導出できる．

1. では「場合分け」を，2. では「帰納法」を用いて証明されている．1. は，推論規則 P-ZERO により，n の値関係なく成り立つことが分かるので，任意のペアノ自然数 n の場合のみ，つまり 1 通りのみの場合分けで証明することができる．一方 2. では，n の値によって，導出の形が異なる．例えば，$n = \texttt{S(Z)}$ の場合，推論規則 P-SUCC を 1 回，$n = \texttt{S(S(S(Z)))}$ の場合，推論規則 P-SUCC を 3 回用いると導出できる．よって，1. のように任意の値 n に対してのみでは成り立つと言えないため，「帰納法」を用いる．帰納法とは，個別的な事例から共通の性質を見出し，法則を導き出す考え方である．2. の証明では，$n \texttt{ plus Z is } n$ を導出するためには，$\texttt{Z plus Z is Z}$ から，推論規則 P-SUCC を n の値の回数分適用することで導出できる，という法則を数学的帰納法を用いて証明している．

5　まとめと今後の課題

本研究の証明で用いる導出システムについて理解を深めた．また、証明法の比較により，構造帰納法について理解を深めた．今後は，導出に関する帰納法を用いたプログラムの性質の証明を行うとともに，構造帰納法と導出に関する帰納法を比較し，導出に関する帰納法について理解を深める．また，プログラムの性質を証明するには，数学的帰納法や累積帰納法，構造帰納法，導出に関する帰納法などを間接的に用いることは可能だが，それぞれのプログラムの証明の性質を直接的に用いるのはどの帰納法かを確認する．さらに，様々な帰納法が整礎帰納法の特殊ケースになっていることを確認する．

参考文献

[1] 五十嵐淳，プログラミング言語の基礎概念，サイエンス社 (2020)