

拡張TSチャートによる 5R-7 Prologの記述について

シナガ メンジャミン、大原 茂之、飯田 昌盛

東海大学

1. はじめに

Prologを図的に表現する手段として、AND-ORツリーがある。しかし、この図的表現では、階層構造、制御構造などを表現することができなかった。また、ユニフィケーションやカットオペレータの概念等が手続き型言語にないため、手続き型言語なれた者にとってPrologを自由に使えるようになる上での障壁となっていた。このような問題点を解決するために、本報告では、Prolog言語の実行順序、AND、OR、カットオペレータの動作をチャートで表現する方法および、そのチャートからのコーディング方法について述べる。

2. 本文

2.1 記号体系

使用する記号体系は、拡張TSチャート¹⁾のうち、以下のものである。

『処理記号』： モジュール記号、マクロモジュール記号、外部変数記号

『接続記号』： ユニフィケーションアーク記号、ライトアーク記号、ステータスアーク記号

『制御記号』： AND制御記号、OR制御記号、カット制御記号

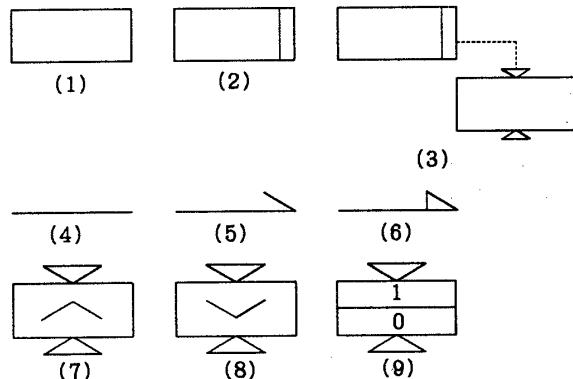


図1 拡張TSチャート記号

【定義1】図1(1)の記号をモジュール記号という。モジュール記号は、処理あるいは機能である。

【定義2】図1(2)の記号を、マクロモジュール記号という。マクロモジュール記号は内部が拡張TSチャートで定義される記号である。

【定義3】図1(3)の記号bをマクロモジュール記号hに対する詳細モジュールという。

【定義4】図1(4)の記号をユニフィケーションアーク記号という。ユニフィケーションアーク記号は、

モジュール記号間でのデータの取り込みを定義する。

【定義5】図1(5)の記号をライトアーク記号という。ライトアーク記号は、処理記号間でのデータの受渡しを定義する。

【定義6】図1(6)の記号をステータスアーク記号という。ステータスアーク記号は、処理記号間でのステータスの受渡しを定義する。

【定義7】図1(7)の記号をAND制御記号という。AND制御記号は、入力すべてが成功ならば、出力を成功にする。入力が一つでも失敗であるならば、出力を失敗にする。

【定義8】図1(8)の記号をOR制御記号という。OR制御記号は、OR制御入力の内、一つでも入力が成功ならば、出力を成功にする。すべての入力が失敗ならば、出力を失敗にする。

【定義9】図1(9)の記号をカット制御記号という。カット制御記号は、カットオペレータに対応する記号である。

2.2 拡張TSチャートとProlog言語

ここでは、拡張TSチャートをProlog言語へコーディングするために、各記号をどのように解釈するかを定義する。

【解釈1】モジュールの内部仕様は、Prologの述語とし、モジュール名は、述語名に引数を加えたものとする。モジュールへの入力は述語の引き数に対応させ、モジュールの出力は述語のリターン値に対応させる。

【解釈2】マクロモジュールの内部仕様は、Prologの節とし、マクロモジュール名は、節の頭部の述語名に引数を加えたものとする。マクロモジュールへの入力は頭部の引数に対応させ、マクロモジュールの出力は頭部のリターン値に対応させる。

【解釈3】外部変数の内部仕様は、Prologの事実の集合とし、外部変数名は、述語名に引数を加えたものとする。

【解釈4】ライトアーク記号は、モジュールの出力が成功か失敗であるかを受け渡すことと解釈する。

【解釈5】ステータスアーク記号は、詳細モジュール間で実行順序や制御の流れであると解釈する。

2.3 モジュール間の接続

ここでは、モジュール間の接続について述べる。

【定義10】接続(m_1, a, m_2)において、 m_1 を先行モジュールと呼び、 m_2 を後続モジュールと呼ぶ。

【規則1】接続 (m_1, a, m_2) において、 a が、ユニフィケーションアーク記号である場合、ユニフィケーションアーク記号は、処理記号 m_1, m_2 が表す述語に適当な値をユニフィケーションすることを表す。ユニフィケーションアーク記号の接続例を図2(1)に示す。

【規則2】接続 (m_1, a, c) において、 c が、AND制御記号あるいはOR制御記号である場合、 a はライトアーク記号である。ライトアーク記号の接続例を図2(2)に示す。

【規則3】接続 (m_1, a, m_2) において、 a が m_2 の上端へ接続し、かつ、どちらかの m_2 がカット制御記号に出力するならば、他の m_2 の上端への入力は、ステータスアーク記号で接続する。カット制御記号の接続例を図2(3)に示す。

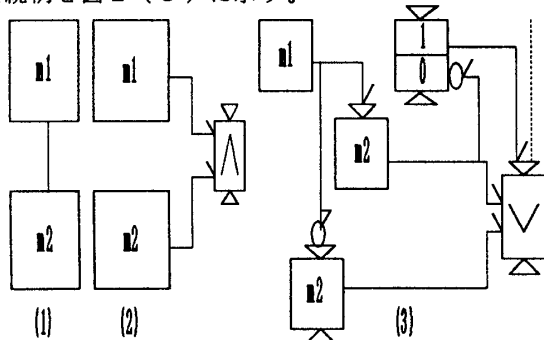


図2 記号の接続

2.4 コーディングの例

ここでは、図3に示すチャートをPrologへコーディングする例を示す。

図3のチャートは、次のような意味を持つ。

$a(X, Y)$ 、 $c(Z, Y)$ はマクロモジュールである。 $b(X, Z)$ 、 $c(Z, Y)$ は、マクロモジュール $a(X, Y)$ に対する詳細モジュールである。ここで、マクロモジュール $c(Z, Y)$ は、二つの詳細モジュールで定義されている。一方は、 $d(Z, M)$ 、 $!$ 、 $e(M, N)$ 、 $f(N, Y)$ 、そして他方は $g(Z, Y)$ である。また b 、 d 、 e 、 f 、 g は、外部変数で、事実の集合である。外部変数は、各モジュールに対して外部にある。

モジュール $b(X, Z)$ が成功すると、AND制御が実行中になり、処理はマクロモジュール $c(Z, Y)$ に移る。マクロモジュール $c(Z, Y)$ は、詳細モジュール $d(Z, M)$ 、 $!$ 、 $e(M, N)$ 、 $f(N, Y)$ や $g(Z, Y)$ から成っている。最初に、モジュール $d(Z, M)$ を実行する。モジュール $d(Z, M)$ が失敗ならば、モジュール $g(Z, Y)$ を実行する。成功ならば、カット制御を実行し、OR制御を制御する。次に、モジュール $e(M, N)$ 、 $f(N, Y)$ を実行する。各モジュールの出力はAND制御で判断し実行される。各詳細モジュールの処理を終了すると出力はOR制御で判断し実行される。OR制御は、マクロモジュール $c(Z, Y)$ の出力になる。モジュール b (

X, Z) とマクロモジュール $c(Z, Y)$ の出力がともにそろってAND制御で判断し実行される。それがモジュール $a(X, Y)$ の出力となる。

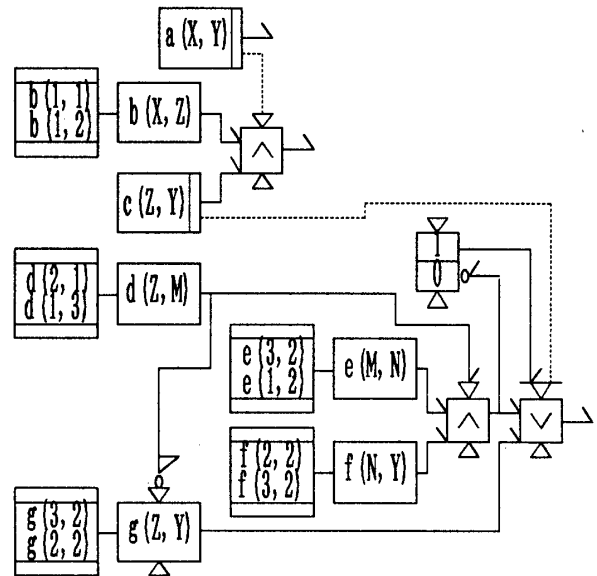


図3 拡張TSチャートの例

図3をProlog言語へコーディングした結果を以下に示す。

```
a(X, Y) :- b(X, Z), c(Z, Y).
c(Z, Y) :- d(Z, M), !, e(M, N), f(N, Y).
c(Z, Y) :- g(Z, Y).
```

```
b(1, 1). | e(1, 2).
b(1, 2). | f(2, 2).
```

```
d(2, 1). | f(3, 2).
d(1, 3). | g(3, 2).
```

```
e(3, 2). | g(2, 2).
```

3. おわりに

本報告では、AND制御記号、OR制御記号、カット制御記号、ユニフィケーションアーク記号によるProlog言語の図的表現とコーディング方法を示した。

謝辞 本研究を進めるに当たり、多くの面でご援助いただいた、本学工学部長萩三二教授、制御工学科小高明夫主任教授ならびに研究室の皆様へ感謝の意を表します。

参考文献

- 1) 安齊、大原、小高：拡張TSチャートにおけるオブジェクト指向言語表現への一考察、情報処理学会第40回全国大会、4S-6、1990
- 2) 安齊、大原、小高：拡張TSチャートにおけるオブジェクト指向言語表現への一考察(2)、情報処理学会第41回全国大会、1H-8、1990