

7S-4

仕様記述言語 LOTOS の図式表現

佐藤 文明, 勝山 光太郎, 水野 忠則

三菱電機(株)情報電子研究所

1. はじめに

通信プロトコルの形式的記述言語として, Estelle¹⁾, LOTOS²⁾, 及びSDL³⁾がISOとCCITTによって標準化されている. 特に, LOTOSは, 時間順序の論理と抽象データ型に基づく仕様記述言語であり, 抽象度が高く, 現在研究が盛んに行なわれている. また, 実際のプロトコルをLOTOSで記述する作業が行なわれている.

しかし, 抽象度が高いことから, SDLやEstelleと比べると, 直感的な理解性が不足しているといわれている. そのため, ISOでは, LOTOSの図式表現であるG-LOTOSの標準化作業に着手している. しかし, 現在は, その図式表現も動作部に限定されており, タイプ定義の部分に関してはまだ検討がなされていない. ここでは, LOTOSのタイプ定義における図式表現について検討し, 記述を行なってみた結果について報告する.

2. LOTOSの図式表現

現在ISOで規定しようとしているLOTOSの図式表現は, 作業の順番として動作定義の部分に限定して検討を開始している. その検討方針としては,

- (1) LOTOS自体の意味定義を変更するものではない.
- (2) 図式は, 手書きでも書きやすく, 計算機でも支援しやすいものとする.
- (3) 動作イベントの選択, 並列動作, 順序性が視覚的に訴えるものとする.
- (4) 図式表現とテキスト表現は, 混合記述ができるものとする.

である. 図1にLOTOSの図式表現の一部を記す. これらの図式表現は, テキスト表現と1対1の対応をなすように考慮されている. しかし, 図式表現は, 一般にテキスト表現よりも多くの情報を持ったために, 1対1に完全に相互変換できるも

のではない. つまり, 図式表現Aからテキスト表現Bに変換し, テキスト表現Bを更に図式表現Cに逆変換しても, AとCが完全に一致する保証はできない.

また, G-LOTOSでは, LOTOSの仕様の大きな部分を占めるタイプ定義に関する図式表現をまだ持っていない. 現在のところ, タイプ定義をいれる箱を用意する予定である. しかし, LOTOSの仕様の理解性の不足に関しては, 抽象データ型に基づいているタイプ定義の問題が大きく, タイプ定義の図式表現は, LOTOSの理解性の向上に影響を与えると考えられる.

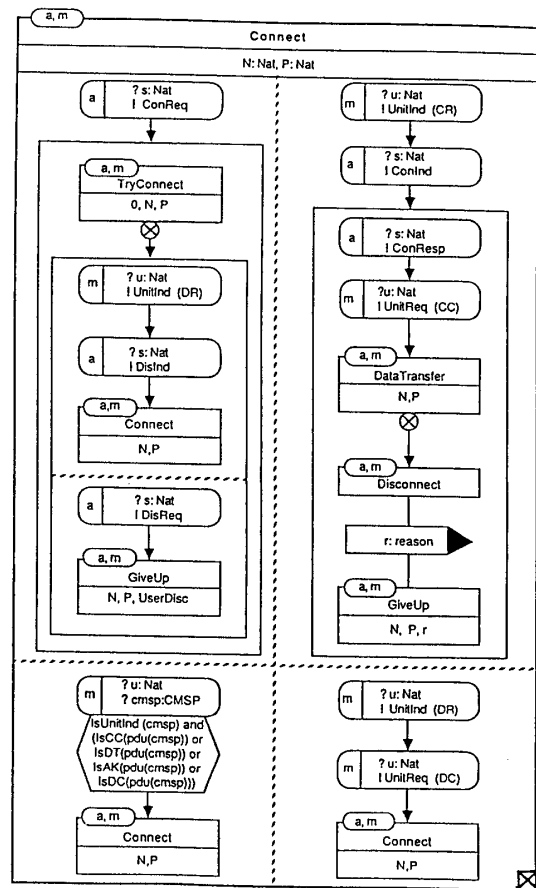


図1. LOTOSの図式表記例

3. データタイプの記述

LOTOSのタイプ定義は、抽象データ型のACT ONEという定義方法を用いている。これは、タイプを代数的に定義するもので、ソート(型)、ソートに定義される演算、及び演算の動作を決定する等式を用いて定義する。図2にタイプ定義の例を示す。

また、LOTOSでは、このように直接タイプを定義する機能の他に、タイプを継承する機能、パラメータ化する機能、他のタイプの名前を換えて再利用する機能がある。

4. データタイプの図式表現

ここでは、先に述べたISOの方針と、以下の方針に基づいてデータタイプの図式表現を提案した。

- (1) タイプ定義は、大きな定義となりやすいため、分割記述ができるようにする。
- (2) タイプ定義は、記述する内容が項目ごとに分かれているため、構造化した記述をとりやすい。このような性質を表現できるようにする。
- (3) タイプ定義の中でも、特に演算の図式化を中心に検討する。

図3に、図2で記述したタイプ定義の図式表現を示す。図3の中で、楕円で記述された部分がソートを表わし、矩形でかこまれたものが演算を表わしている。演算とソートは、その関係が線で連結されることで表現されている。

5. 考察

上記のようなタイプ定義の図式表現を使って、実際のLOTOS仕様を図式表記した結果、以下

```

type Nat is
  sorts nat
  opns 0:->nat
       succ:nat ->nat
       _+_ :nat,nat->nat
  eqns of sort nat
    forall x,y:nat
      x+0 = x;
      x+succ(y) = succ(x+y);
endtype

```

図2. LOTOSのタイプ定義

のような結論が得られた。

(1) ソートの数が少ない場合(5~6個程度まで)は、直感的に分かりやすい記述ができた。しかし、ソート数が多くなると、演算とソートを連結する線の数が多くなり、図式が複雑になって、逆に理解性を損なってしまう場合がある。

(2) 図式を分割する機能は、大きな仕様を図式表記するときには、どうしても必要になる機能である。SDLにおいても、コネクタが存在しており、その意義は大きいものがある。

(3) 構造的な記述を用意しておくことは、記述量や記述スペースに応じて、図式の表記スタイルを選ぶことができるため、有効である。

6. まとめ

仕様記述言語LOTOSのデータタイプ定義に関する図式表現について検討した。現在、ISOでは、G-LOTOSの構文を検討すると同時に、図式言語の構文定義方法の検討も行なっている。今後は、ここで述べたタイプ定義の図式に関して改良を加えていくと同時に、正確な構文定義を検討していく予定である。

[参考文献]

- 1) ISO:Information Processing Systems-Open Systems Interconnection-Estelle-A Formal Description Technique Based on an Extended State Transition Model, ISO 9074(1989).
- 2) ISO:Information Processing Systems-Open Systems Interconnection-LOTOS-A Formal Description Technique Based on the Temporal Ordering of Observational Behavior,ISO 8807(1989).
- 3) CCITT:Functional Specification and Description Language(SDL), Recommendations Z.100(1988).

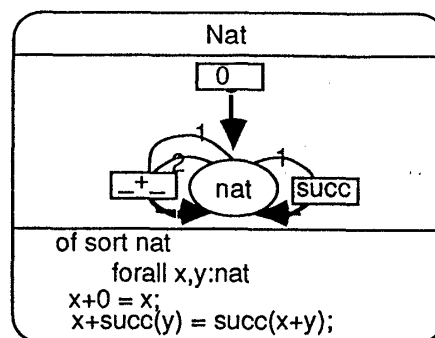


図3. LOTOSタイプ定義の図式表記