

AESOPにおけるプロトタイピング手法

4M-11

種田 克行 笠原 匡孝 唐沢 圭 岩田 誠 寺田 浩詔

大阪大学 工学部 情報システム工学科

1 はじめに

本研究は、データ駆動パラダイムによる多面的図的仕様記述環境 AESOP(Advanced Environment of Software Production)[1,2,3]において、システム発注者や設計者の意図をソフトウェア仕様記述に正確に反映させるためのプロトタイピング環境の構築を目的としている。

本稿では、データ駆動原理の検証性を活用して、加法的に仕様を完結させる核型プロトタイピングをソフトウェア仕様記述水準で可能にする手法を提案する。

2 AESOP のプロトタイピング機能

プロトタイピングは、実際のプログラムの実行動作の結果や状況を通して、記述者の意図と実行可能プログラムが合致しているかどうかを確認する有効な手段の一つである。AESOP ではこのプロトタイピングを効果的に行なえる環境を提供するために、ソフトウェア仕様を記述する過程で、部分的にでも仕様が完結してデータ駆動的に解釈実行可能になれば、いつでも、プロトタイプを仕様記述水準で視覚的に試用できる機能を提供しようとしている。

本研究ではこの AESOP 環境において、半形式的で完結していない過渡的なソフトウェア仕様記述を含めた広範囲の AESOP 仕様記述を対象として、スタブを埋め込んででもプロトタイプ実行を可能にする手法を検討した。

2.1 プロトタイピング可能な仕様

厳密なデータ駆動原理に従えば、ある機能モジュールは、全ての入力アーク上にデータが揃った時、その機能を実行でき、その実行結果を全ての出力アーク上へのデータの生成として反映する。しかしながら、機能モジュールの実行可能性をより柔軟に捉えれば、ある特定の入力の組に因って、着目している一つの出力アーク上にデータが生成できれば、その機能モジュールが充分実行可能であるとみなせる。

AESOP 仕様記述からは、データ依存グラフ情報に加えて、機能モジュールの入出力データの因果関係の情報や入出力データ構造の情報が抽出できるので、先

のように実行可能性を捉えても、その判断を半自動的に行なえる可能性がある。

したがって本研究では、AESOP 図的仕様記述から本質的な情報を抽出する相互変換機能 [3] を介して生成される、抽象データ駆動型プログラム情報 [2] 上で実行可能なプログラム仕様を以下のように定義し、そして、この水準で、実行可能な部分、および、スタブの埋め込みにより実行可能になりうる部分を検出して、プロトタイプとして生成する手法を採用した。

定義1 (プロトタイピング可能な仕様): AESOP 仕様記述から生成された抽象データ駆動型プログラム情報を S 、その入力ポート i_{S_j} ・出力ポート o_{S_j} の集合をそれぞれ $I(S)$ 、 $O(S)$ とすると、

$$k > 0, \exists o_S \in O(S), \exists i_{S_1}, \exists i_{S_2}, \dots, \exists i_{S_k} \in I(S);$$

$$o_S \stackrel{P}{\Rightarrow} i_{S_1}, i_{S_2}, \dots, i_{S_k}$$

である時、 S を定義している仕様記述はプロトタイピング可能な仕様である。ただし、 \Rightarrow はある入力トークンの組によりある出力トークンが、 S に含まれるモジュール M に関する因果関係の変換規則 P を連鎖的に適用して生成されることを示す。ここで P は、

$$P = \{o_M \Rightarrow I_{o_M}(M) \mid M \in S, o_M \in O(M), I_{o_M}(M) \subseteq I(M)\}$$

と定義される規則である。 $I_{o_M}(M)$ は、 M に関して、 o_M を生成するための入出力因果関係と入出力シーケンスを持つ入力ポート集合である (図.1)。 □

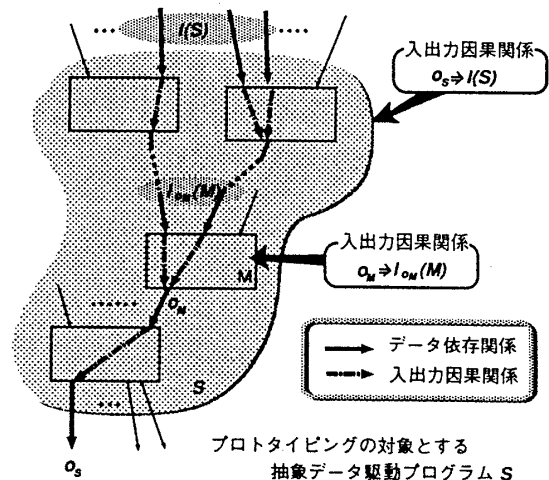


図1: 抽象データ駆動型プログラム水準での実行可能性

A Study on Prototyping Scheme for AESOP.
Katsuyuki TANEDA, Masataka KASAHARA, Kei KARASAWA,
Makoto IWATA, Hiroaki TERADA.
Department of Information Systems Engineering, Faculty of Engineering,
OSAKA University

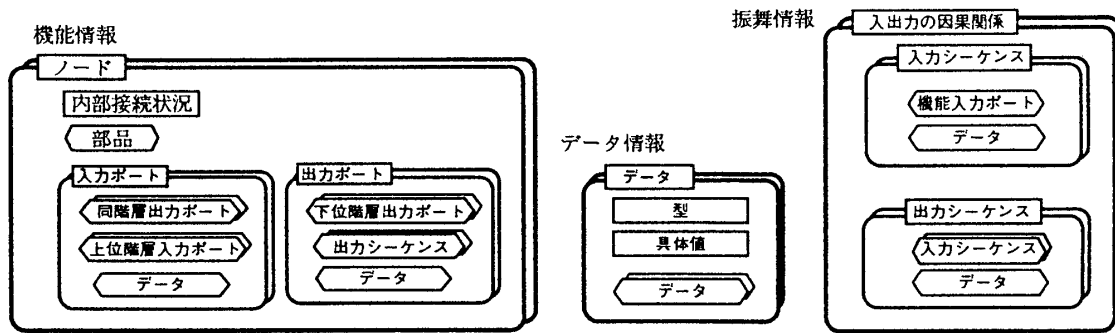


図 2: プロトタイピング機能で用いるデータの構成例

2.2 プロトタイピング可能な仕様への誘導手法

半形式的で完結していない過渡的なソフトウェア仕様記述はプロトタイピング可能でないことが多い。仕様記述が最終的にはデータ駆動型に動作するプログラム仕様に素直に帰着できる事実を考慮すれば、このような完結していない機能モジュールをデータ駆動型に仮に動作させて実行を試みることで、核型プロトタイピングの効果をより増大できることが判る。したがって、AESOP のプロトタイピング機能では、次のような機能を記述者に提供している。

- i). 定義 1 を満足する部分プログラムを検出し記述者に提示する。
- ii). 完結していない機能モジュールを仮にデータ駆動的に解釈した場合に、定義 1 を満足する部分プログラムの検出と提示を行なう。
- iii). これらの部分プログラムに属する機能モジュールの中で、機能(下位階層あるいは部品)が未確定のものについて、入出力の因果関係とシーケンスを元にデータ駆動型スタブを生成して埋め込み、実行可能プログラムを生成する。

このように積極的にプロトタイピング実行を許すことによって、プロトタイプ可能な仕様記述を完成する方向に記述者を誘導することが可能になる。

3 プロトタイピング機能の一構成法

前述のような核型プロトタイピング機能の効果的な実現には、多面的かつ階層的な仕様記述の追加・修正に対して、スタブ埋め込みの鍵となる機能モジュールの入出力の因果関係とシーケンスを常に更新し、抽象データ駆動型プログラムへの変換ならびにその検証がインクリメンタル(加法的)かつ容易に実現できることが重要となる。

定義 1 は、データ構造を鍵とした機能モジュール間のデータ依存関係、そのモジュールの入出力因果関係と入出力シーケンスの検査と捉えることができる。そ

のため、対象とする抽象データ駆動型プログラム情報をデータ依存関係を表す機能情報、その入出力の因果関係とシーケンスを表す振舞い情報、モジュール間で授受されるデータ情報から構成している。図 2 は出力データの生成に必要な入力集合を容易に探索できるようなデータ構成の例である。このデータ構成では、図 2 の右に示すように、前述のスタブを容易に生成するために、入出力の因果関係を一群の情報として捉えられるビューも用意している。さらに、実行可能な部分プログラムの検出のための探索空間を極小化するために、検証済みの機能モジュールや部品をブラックボックス化できるよう構成されている。

4 おわりに

AESOP でのプロトタイピングは、モジュールの入出力因果関係とシーケンス、データ構造を考慮すると、単純なデータフローグラフを対象とするプロトタイピングよりも広いクラスに属するプログラムを検証できる。

本稿ではその特徴を利用し、ソフトウェア仕様記述からスタブを埋め込んで部分的にデータ駆動実行可能なプログラムを生成する手法の素案を提案した。

今後は、プロトタイピング可能な範囲の検出手法として、より高位の入出力因果関係、特にモジュール間の入出力シーケンスの整合性を加味した検出手法を追究することが残されている。

謝辞 本研究に関し御指導・御支援頂きました大阪大学工学部寺田研究室の各位に深く感謝致します。なお、本研究の一部は、文部省科研費(一般 B-2 05452363, 試験 B-1 06555110)の援助によるものである。

参考文献

- [1]. 西川、寺田 他:“超高位図的仕様記述環境(AESOP)の構想”, 情報処理学会計算機アーキテクチャ研究会, 90-ARC-83-2, pp.7-12 (1990-07).
- [2]. 岩田、寺田:“図的仕様記述からのデータ駆動プログラムの生成手法”, 情報処理学会計算機アーキテクチャ研究会, 94-ARC-107-8, pp.57-64 (1994).
- [3]. 唐沢 他:“AESOP における仕様記述相互間の変換手法”, 第 49 回情報処理学会全国大会, 4M-10 (1994).