7L-8

ＤＦ型ＴＳチャートで記述されたプログラムの並列性の抽出

孫 志太、 大原 茂之、 飯田 昌盛
東海大学

１．はじめに

ＤＦ型ＴＳチャートで並列処理システムを記述することができる(1),(2)。しかし、ＤＦ型ＴＳチャートをソフトウェア支援システムに組み込んで、並列処理プログラムを自動生成するとなると、いくつかの問題がある。たとえば、モジュールの実行順序の決定、並列動作可能なモジュール群の抽出などである。本報告は、これらの問題を解決する一つの試みとして、ＤＦ型ＴＳチャートを、マトリクス化手法を利用して並列性を抽出する手順を示す。

２．本文

２．１　データの流れとモジュールの関係

ＤＦ型ＴＳチャートでは、モジュール間を入出力アークやリードアークなど、意味づけしたアークで接続する(3)。まず、入出力アークとリードアークでモジュール間を接続したＤＦ型ＴＳチャートについて述べる。ただし、外部変数モジュールは、ｆ（ｘ）＝ｘ なる処理モジュールと見なす。

【定義１】ＤＦ型ＴＳチャートＴに対し、Ｔを構成するモジュールの集合をＭ（Ｔ）と記す。Ｍ（Ｔ）において、モジュールｍの実行タイミングよりも先に実行しなければならないモジュールを、ｍの先行モジュールといい、この集合をＰ（ｍ）と記す。反対に、ｍの実行タイミングよりも後に実行しなければならないモジュールを、ｍの後続モジュールといい、この集合をＲ（ｍ）と記す。

【定義２】ｗ，ｖ∈Ｍ（Ｔ），ｗ≠ｖに対し、ｗとｖが互いに任意のタイミングで実行できるならば、ｗとｖは並列実行可能であるという。

【補題１】∃ｗ，∃ｖ∈Ｍ（Ｔ）に対し、
ｗ∉Ｐ（ｖ）∧ｗ∉Ｒ（ｖ）　なる必要十分条件は、　ｖ∉Ｐ（ｗ）∧ｖ∉Ｒ（ｗ）である。

［証明］必要性。∃ｗ，∃ｖ∈Ｍ（Ｔ）に対し、ｗ∉Ｐ（ｖ）∧ｗ∉Ｒ（ｖ）ならば、定義１より、ｖ∉Ｒ（ｗ）とｖ∉Ｐ（ｗ）が得られる。故に、ｖ∉Ｐ（ｗ）∧ｖ∉Ｒ（ｗ）である。十分性も同様にして証明できる。

補題１と定義２より以下の定理を得る。

【定理１】∃ｗ，∃ｖ∈Ｍ（Ｔ）に対し、
ｗ∉Ｐ（ｖ）∧ｗ∉Ｒ（ｖ）ならば、ｗとｖは並列実行可能である。

２．２　ＤＦ型ＴＳチャートのマトリクス化

ここでは、ＤＦ型ＴＳチャートを、モジュールを頂点とする有向グラフとして扱い、ＤＦ型ＴＳチャートをマトリクスで表す。

【定義３】ＤＦ型ＴＳチャートＴにおいて、ｗ，ｖ∈Ｍ（Ｔ）に対し、ｗとｖを端点とする入出力アーク、または、リードアークが存在するならば、ｗとｖは互いに隣接しているという。

定義３に基づいて、ＤＦ型ＴＳチャートから、モジュールの隣接マトリクスを作る規則を定める。

（規則１）Ｍ（Ｔ）に属するｎ個のモジュールに、１から順に番号を付ける。

（規則２）ｎ×ｎマトリクスＡの各要素Ａijは、モジュールｍiとｍjの間の関係を表す。Ａij（１≦ｉ≦ｎ、１≦ｊ≦ｎ）の内容を表１で示すように決める。

表１　Ａijの決定

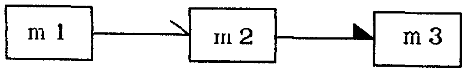
ｍiとｍjの関係	Ａij	Ａij
隣接していない、または、ｉ＝ｊ	０	０
ｍiからｍjへ入出力アークで接続	１	－１
ｍiからｍjへリードアークで接続（ｍiの前にまたアークで接続）	１	－１
ｍiからｍjへリードアークで接続（ｍiの前にアークがない）	－１	１

【定義４】規則１～規則２で作ったマトリクスは、初期マトリクスといい、Ａ（０）と記す。

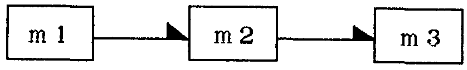
上記規則による記述例を図１に示す。

【定義５】マトリクスＡ（ｃ）（ｃ≧０，Ａ≠０）から、行ベクトルに－１を含まない全てのｍの行ベクトルと列ベクトルを、０ベクトルで書き換えたものを新しいマトリクスＡ（ｃ＋１）とする。

Extraction of Parallelism in DF Type TS Chart
Zhitai SUN,Shigeyuki OHARA,Masamori IIDA.
TOKAI UNIV.



(a)



(b)

$$\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$

(c) (a) のマトリクス　(d) (b) のマトリクス

図1 DF型TSチャートとそのマトリクス表現

【定義6】m∈M（T）に対し、マトリクスA（c）において、P（m）＝φであれば、mをA（c）における始端モジュールという。A（0）における始端モジュールの集合をS（m）と記す。R（m）＝φになるモジュールmを終端モジュールといい、終端モジュールの集合をT（m）と記す。

【定理2】mの行ベクトルが－1の元を含まないならば、mは始端モジュールである。

［証明］マトリクス化規則により、mはその行ベクトルが－1の元を含まないならば、先行モジュールは存在しない。故に、定義6よりmが始端モジュールである。

【定義7】マトリクスA（c）における始端モジュールの集合に対し、cをそれらの実行クラスと呼ぶ。実行クラスcであるモジュールの集合をE（c）と記す。

2．3　並列実行可能モジュール群の抽出

初期マトリクスに基づき、モジュールの実行クラスを決めることと、各モジュールの先行モジュールの集合を求めることにより、並列実行可能モジュール群の抽出を行う。その抽出手順を次に示す。

《手順1》

〔ステップ1〕初期化する。クラスcを0とし、現マトリクスを初期マトリクスとする。

〔ステップ2〕定義1と定理2に基づいて、現マトリクスにおける始端モジュールを捜し出し、実行モジュールの集合E（c）に書き入れる。

〔ステップ3〕∀m∈E（c）に対して、それぞれの行ベクトルと列ベクトルを0で書換え、新しいマトリクスを作る。

〔ステップ4〕クラスcをインクリメントする。

〔ステップ5〕A（c）＝0になるまで、ステップ2～ステップ4を繰り返して実行する。

〔ステップ6〕終了する。

【定理3】同じ実行クラスを有するモジュールは、並列実行可能である。

［証明］w，v∈M（T）に対して、同じ実行クラスcを有するとする。定義7より、両方ともcクラスのマトリクスにおける始端モジュールである。定義2と定義6より、wとvは並列実行可能である。

違う実行クラスを有するモジュールに、並列実行可能なものを捜し出す手順を次に示す。

《手順2》

〔ステップ1〕∀mi∈M（T）に対して、初期マトリクスA（0）から、miの先行モジュールの集合P（mi）を、次式で求める。

P（mi）＝{mj | Aij＝－1}

〔ステップ2〕1≦i≦n，1≦j≦nに対して、mj∈P（mi）ならば、次の計算を行う。

P（mi）＝P（mi）∨P（mj）

〔ステップ3〕1≦i≦n，1≦j≦nに対して、

{mi∉P（mj）∧mj∉P（mi）}

∧{P（mi）∩P（mj）＝φ}ならば、

miとmjは並列実行可能である。

〔ステップ4〕終了する。

3．おわりに

本報告では、入出力アークとリードアークでモジュール間を接続したDF型TSチャートにおけるプログラムの並列性を抽出する手順を示した。この他に、シーケンスアークやコントロールアークなどを含む、DF型TSチャートの並列性を解析することについては、次の機会に報告したい。

謝辞　本研究を進めるに当たり、日頃お世話になっている本学工学部長荻三二教授に感謝の意を表します。

参考文献

(1)大原茂之：木構造化チャートによるプログラム開発・保守技法、情報処理学会論文誌、Vol.27 No.10 ,1986.

(2)S.OHARA:The Modelling and Analysis of Concurrent Systems by Tree Structured Charts,Proceedings of th 11th Joint Symposium TOKAI UNIV .-Technichal UNIV.BUDAPEST,1987.

(3)相浦，大原，小高：ソフト開発用データフローチャートとそのコーディング手順について、第36回情報処理全国大会3L-2,1988.