

氏名	にし 西	たに 谷	やす 泰	あき 昭
授与学位	工 学 博 士			
学位授与年月日	昭和 58 年 5 月 11 日			
学位授与の根拠法規	学位規則第 5 条第 1 項			
研究科, 専攻の名称	東北大学大学院工学研究科 (博士課程) 情報工学専攻			
学位論文題目	並行プロセスの記述と同期に関する理論的研究			
指導教官	東北大学教授 伊藤 貴康			
論文審査委員	東北大学教授	伊藤 貴康	東北大学教授	斎藤 伸自
	東北大学教授	木村 正行	東北大学教授	野口 正一

論 文 内 容 要 旨

第 1 章 序 論

計算機の処理において、複数のプロセスが相互作用を行いつつ仕事を進める応用は、オペレーティングシステム、リアルタイム処理、コンピュータネットワークなど少なくない。これらの分野では基本的概念として、並行プロセスと同期機構の概念が重要であり、その研究も数多くなされている。しかし、並行プロセスと同期機構の形式的枠組、その枠組における解析など理論的研究は十分になされているとは言えない。

本論文の目的は、並行プロセスと同期機構の理論的な研究である。そのための形式的なモデルとして、並行プロセスと同期機構を形式言語で記述するフロー表現モデルを導入し、言語理論的な観点から解析を行う。そして、フロー表現モデルの記述能力、同期機構の能力、他の並行プロセスモデルとの関係などを明らかにするとともに、並行プロセスの同期問題、デッドロック問題について言語理論の立場から論じる。

第 2 章 並行プロセスのフロー表現モデルによる記述

本章では、フロー表現モデルの定義を与えるとともに、フロー表現モデルによる並行プロセスの記述例を示すことによって、モデルの有用性を示している。フロー表現モデルは、並行表現を基本として、並行プロセスの制御の流れ（計算系列）を記述するモデルである。

並行表現は、正規表現にシャッフル演算子 \parallel とその閉包 $\textcircled{\parallel}$ を加えて拡張した表現であり、その意味は正規表現と同様に言語で与えられる。 $E = E_1 \parallel E_2$ の記述する言語 $|E|$ は、 $\{x_1 y_1 x_2 y_2 \cdots x_k y_k \mid x_i, y_i \in \Delta^*, x_1 x_2 \cdots x_k \in |E_1|, y_1 y_2 \cdots y_k \in |E_2|\}$ であり、 $E^{\textcircled{\parallel}} = \lambda + E + E \parallel E + \cdots$ である。並行表現は、並行プロセスの並行性をシャッフル演算子で簡潔に記述できるが、同期を含めた記述は難しい。そこで、同期を含めた並行プロセス記述のモデルとしてフロー表現モデルを与える。

フロー表現モデルは、プロセス記述 E と同期機構 K の 2 項組 $\langle E, K \rangle$ で与えられる。ここで、 E は $\Sigma \cup \Gamma$ ($\Sigma \cap \Gamma = \emptyset$, Σ は原子記号集合, Γ は同期記号集合) 上の並行表現であり、 K は Γ 上の言語である。フロー表現モデル $\langle E, K \rangle$ が記述する言語 $L_K(E)$ は、 $\{h(x) \mid x \in |E|, \bar{h}(x) \in K\}$ である。ここで、 h, \bar{h} は準同形写像であり、それぞれ、 Σ の元、 Γ の元だけを取り出す。

プロセス記述 E を正規表現に制限したフロー表現モデルを、正規表現モデルと言う。正規表現モデルはペトリネットや文脈自由文法と密接な関係があり、重要なモデルである。

フロー表現モデルの定義では、同期機構 K について何ら制限を加えていないが、本論文では代表的な同期機構として、5 種類のセマフォ機構と Dyck 機構を考える。セマフォ機構は、通常、セマフォと呼ばれる同期変数に対する Wait/Signal 操作によって定義されるが、本論文ではフロー表現モデルの定義に従って、操作系列の集合として次のように定義する。なお、 n はセマフォの個数を表わしている。

$$\Gamma_n = \{\sigma_i, \omega_i \mid 1 \leq i \leq n\}$$

- (1) カウンタセマフォ機構 $C(n) = |(\sigma_1 \omega_1 + \sigma_1)^{\textcircled{\parallel}} \parallel (\sigma_2 \omega_2 + \sigma_2)^{\textcircled{\parallel}} \parallel \cdots \parallel (\sigma_n \omega_n + \sigma_n)^{\textcircled{\parallel}}|$
- (2) $[0]$ -カウンタセマフォ機構 $C_0(n) = |(\sigma_1 \omega_1)^{\textcircled{\parallel}} \parallel (\sigma_2 \omega_2)^{\textcircled{\parallel}} \parallel \cdots \parallel (\sigma_n \omega_n)^{\textcircled{\parallel}}|$
- (3) 2 値セマフォ機構 $B(n) = |(\sigma_1 \omega_1 + \sigma_1)^* \parallel (\sigma_2 \omega_2 + \sigma_2)^* \parallel \cdots \parallel (\sigma_n \omega_n + \sigma_n)^*|$
- (4) $[0]$ -2 値セマフォ機構 $B_0(n) = |(\sigma_1 \omega_1)^* \parallel (\sigma_2 \omega_2)^* \parallel \cdots \parallel (\sigma_n \omega_n)^*|$
- (5) 単純 $[0]$ -2 値セマフォ機構 $SB_0(n) = |(\sigma_1 \omega_1 + \sigma_2 \omega_2 + \cdots + \sigma_n \omega_n)^*|$

Dyck 機構 $D(n)$ は、 Γ_n 上の Dyck 言語として定義される (Dyck 言語は n 種類の括弧が整合している括弧系列の集合である)。

第 3 章 フロー表現モデルの記述能力と同期機構

本章では、フロー表現モデルの記述能力を調べることによって、セマフォ機構および Dyck 機構の同期機構としての能力を調べている。

まず、同期機構として並行表現で記述できるものを考えたとき、フロー表現モデルおよび正規表現モデルが帰納的可算集合を記述できることを示した。そして、セマフォ機構および Dyck 機構が、並行表現で記述された任意の同期機構をシミュレートできるという意味で万能であることを示した。この結果は、セマフォ機構および Dyck 機構がフロー表現モデルの同期機構として十分な能力を持ち、互いに等しい能力を持つことを示している。

次にセマフォの個数に注目して、セマフォ機構および Dyck 機構が 3 個のセマフォを持てば、並行表現で記述された任意の同期機構をシミュレートできること、すなわち万能であることを示した。

この結果は、わずか4状態の有限オートマトンで同期機構を構成すれば、十分な同期制御ができることを示す興味深い結果である。

第4章 正規表現モデルの記述能力と同期機構

フロー表現モデルは、同期機構を単純なものに制限しても帰納的可算集合を記述できるが、これは、フロー表現モデルで記述された並行プロセスに関する種々の性質が決定不能であることを示している。そこで本章では、プロセス記述を制限したモデルである正規表現モデルについて、その記述能力と同期機構の能力を調べている。正規表現モデルについて論じる理由は3つある：(1)カウンタセマフォ機構または[0]-カウンタセマフォ機構を持つ正規表現モデルは、ペトリネットと密接な関係がある(第5章)、(2)Dyck機構を持つ正規表現モデルは、計算機科学の分野で重要な言語である文脈自由言語を記述できる(本章)、(3)正規言語は言語理論的に扱い易く、シャッフル演算子 \parallel を導入してもその記述能力が変わらない。

本章で示した正規表現モデルの記述に関する結果は次の通りである。(1)セマフォ機構を持つ場合、その記述能力はセマフォ機構の種類によって差がある。(2)カウンタセマフォ機構を持つ場合、セマフォの個数によってその記述能力に差がある。(3)同期機構がカウンタセマフォ機構または[0]-カウンタセマフォ機構であれば、正規表現モデルの記述する言語の族は文脈自由言語の族と比較不能であり、Dyck機構であれば一致する。(4)Dyck機構を持つ場合、セマフォの個数が2以下であれば、その記述能力はセマフォの個数によって差があり、2以上であれば等しい。

(1)、(2)の結果は、正規表現モデルにおけるセマフォ機構の同期機構としての能力の違いを示すとともに、フロー表現モデルの場合との著しい相違を示している。また、(3)の結果は、カウンタセマフォ機構および[0]-カウンタセマフォ機構とDyck機構が、同期機構として本質的な違いを持っていることを示している。

第5章 フロー表現モデルとペトリネット

ペトリネットは並行プロセスの制御の流れを記述する基本的なモデルであり、多くの分野に適用されるとともに、理論的な研究もなされている。本章では、正規表現モデルおよび制限されたフロー表現モデルがペトリネットと等価な記述能力を持つことを示し、フロー表現モデルとペトリネットの関係を明らかにしている。また、正規表現モデルについての第4章の結果から、ペトリネットについての性質を導いている。

ペトリネットを言語の受理機械としたとき、その受理方式によって数種類の言語が定義されるが、ここでは、ペトリネット言語、準ペトリネット言語と呼ばれる2種類の言語を考えた(この言語はそれぞれ、Petersonの定義したL型、G型の言語に対応している)。そして、カウンタセマフォ機構、[0]-カウンタセマフォ機構を持つ正規表現モデルの記述する言語の族が、それぞれ準ペトリネット言語、ペトリネット言語の族と等しいことを示した。また、 \mathbb{N} の上に*がネストしないような並行表現を、プロセス記述、同期機構として持つフロー表現モデルが、ペトリネット言語と等価であることも示した。これらの結果は、所属問題、空問題などの決定問題が、上述の正規表現モ

デルおよび制限されたフロー表現モデルに対して、可解またはペトリネットの到達可能問題に帰着されることを示しており、並行プロセスに関するある種の問題が可解であるフロー表現モデルのクラスを与えている（第6章）。

また、上述の結果と第4章の結果から、準ペトリネット言語について、(1)ペトリネット言語との記述能力の差、(2)p-節点の個数による記述能力の差、言語演算についての閉包性（ \cup , \cap , \cdot , \parallel , 準同形写像, 準同形写像の逆写像については閉, $*$, 補集合については非閉）を示した。

第6章 並行プロセスの制御問題とフロー表現モデル

本章では、並行プロセスの制御問題を同期問題とデッドロック問題に分け、それらをフロー表現モデルを用いて論じている。

同期問題は、相互排除、実行順序など、与えられた性質（同期条件）を並行プロセスが満たすか否かを判定する同期判定問題と、同期条件を満たすように並行プロセスを構成する同期構成問題に分けられる。本章では、同期条件を、与えられた性質を満たす計算系列の集合、すなわち Σ 上の言語 S とみなすことによって、同期判定問題を包含問題（ $L_K(E) \subseteq S$ ）、空問題（ $L_K(E) \cap S^c = \emptyset$ ）として扱うことができた。そして、同期判定問題が可解であるような正規表現モデルと同期条件を与えた（例えば、カウンタセマフォ機構を持つ正規表現モデルに対して、同期条件 S の補集合が準ペトリネット言語であれば、同期判定問題は可解である）。また、相互排除、実行順序、公正さなどの同期条件が並行表現で簡潔に記述できることを示し、その記述からフロー表現モデルを構成する方法、すなわち同期構成問題の解を与えた。

一方、デッドロック問題については、フロー表現モデルがデッドロックフリーであるか否かを、デッドロック集合 $DL(E, K) = \{x \mid x \in \text{prefix}(\{|E\}), \bar{h}(x) \in \text{prefix}(K), \forall a \in \Sigma \cup \Gamma \{x a \notin \text{prefix}(\{|E\}) \text{ または } \bar{h}(x a) \in \text{prefix}(K)\}\}$ の空問題に帰着させた。そして、デッドロック問題が可解な正規表現モデル（同期機構が正規言語）、デッドロック問題がペトリネットの到達可能問題に帰着されるような正規表現モデル（同期機構がカウンタセマフォ機構または $[0]$ -カウンタセマフォ機構）を与えた。

最後に、本章で与えた同期構成法で 3-dining philosophers 問題の解を与え、その解がデッドロックフリーであることをデッドロック集合の空問題を解くことによって示した。

第7章 結 言

本章では各章で得られた結果をまとめて示した。

審査結果の要旨

計算機システムの計算資源を管理するオペレーティングシステムのもとで、見掛け上同時に実行されるプログラムが並行プロセスと呼ばれる。オペレーティングシステムの管理下にある並行プロセスが計算資源の使用をめぐる競争し合うことがあり、このため適当な同期を取って並行プロセスを実行させることが不可欠となる。計算機システムの大規模化に伴って、並行プロセスとその同期に関する解析・設計の理論が要望され、活発な研究が行われつつある。

著者は、このような観点から、フロー表現モデルと呼ぶ並行プロセスの記述モデルを用いて、並行プロセスとその同期についての理論的研究を行った。本論文はその成果をまとめたもので全編7章よりなる。

第1章は序論である。

第2章では、並行プロセスの記述モデルとして、フロー表現モデルと、これに制限を加えた正規表現モデルを与えている。また、セマフォ変数を用いた同期機構であるセマフォ機構および Dyck 機構が、フロー表現モデルを用いて簡明に記述できることを示している。

第3章では、フロー表現モデルの記述能力をオートマトン理論の手法を用いて解析し、フロー表現モデルが帰納的可算集合を記述できるという意味で万能であることを示している。更に、3個のセマフォ変数のみのセマフォ機構および Dyck 機構によって制限されたフロー表現モデルが万能であるという興味深い結果も与えている。

第4章では、正規表現モデルの記述能力について解析を行っている。正規表現モデルの場合には、セマフォ機構の種類やセマフォ変数の個数によって記述能力に差があることを示している。また、Dyck 機構で制御された正規表現モデルの記述能力は文脈自由文法の記述能力と一致することも示している。

第5章では、フロー表現モデルおよび正規モデルとペトリネットの関係を論じている。ペトリネット言語および準ペトリネット言語と等価な正規表現モデルを与え、第4章の結果を用いて、準ペトリネット言語についてのいくつかの新しい結果を示している。

第6章では、同期条件を満たす並行プロセスの設計法をフロー表現モデルの形式を用いて与え、同期判定問題およびデッドロック判定問題について論じている。

第7章は結言である。

以上要するに、本論文はオペレーティングシステムを構成する並行プロセスと同期機構を、フロー表現モデルによって記述し、それらの性質をオートマトン理論の手法を用いて解析し、ソフトウェア工学にいくつかの新しい知見を加えたもので、情報工学に寄与するところが少なくない。

よって、本論文は工学博士の学位論文として合格と認める。