ソフトウェア基礎論　２５－８
（1988. 5. 26）

ω正則集合と等価な表現能力をもつ時相論理

A Temporal Logic Expressively Equivalent to ω-Regular Set

濱口清治　平石裕実　矢島脩三
Kiyoharu Hamaguchi　Hiromi Hiraishi　Shuzo Yajima

京都大学　工学部
Faculty of Engineering, Kyoto University

あらまし　ＶＬＳＩ技術の発達に伴って論理システムの設計・実現の正しさを、形式的に確認するための形式的検証手法の確立が望まれている。形式的検証では、時間の概念を陽に表現することができることから時相論理が利用されつつある。我々は有限オートマトンを設計の対象と捉え、正則集合を表現できる正則時相論理(ＲＴＬ)を提案し、またその真偽判定を利用した検証手法を示してきた。しかし、ＲＴＬは無限長の系列の性質を記述できず、無限長の系列上の動作と捉えたほうが自然な対象を記述することが困難である。そこで、本稿では設計対象をω有限オートマトンと捉え、ＲＴＬを拡張してω正則集合を表現する能力を持つω正則時相論理（ωＲＴＬ）を提案し、また検証に応用することができる真偽判定が決定可能であることを示す。

Abstract　The establishment of formal verification technique has been strongly desired. For this purpose, we have proposed Regular Temporal Logic (RTL), which is expressively equivalent to regular set, and formal verification method by utilizing its evaluation algorithm. RTL, however, can't express behavior on an infinite sequence, thus, in some cases, it is difficult to describe specification for an object which should be expressed on an infinite sequence. In this report, we propose Omega Regular Temporal Logic (ωRTL), which is expressively equivalent to ω-regular set, and prove that its evaluation problem, which can also be utilized in formal verification, is decidable.

１．まえがき

超大規模集積回路（ＶＬＳＩ）により大規模な論理システムの構築が可能となりつつある現在、論理システムを誤りなく設計・実現して、正しく動作していることの確認ができる論理設計手法の確立が重要になってきている。このため、論理設計のＣＡＤ／ＤＡ、論理シミュレーション等の研究が行われて来ているが、数学的な基盤を持つ形式的仕様記述や形式的検証手法については、未だ実用的な手法が確立されておらず、その研究が重要となってきている。

形式的検証を行う際には、まず設計や仕様を形式的に記述し、次にこの記述に対して形式的な検証手法を適用することになる。形式的な記述手法としては、論理体系を用いる方法があり、従来、組み合わせ回路の記述として命題論理が使われてきたという経緯から、命題論理に時間の概念を表現するための時相演算子を加えた時相論理[1]が、用いられるようになってきている[2][3]。設計対象を有限オートマトンと考えると、記述のために用いる時相論理には、正則集合を表現する能力が必要であるが、従来の時相論理はω正則集合を表現することができない。また、Wolperの提唱した拡張時相論理（ＥＴＬ）[4]はω正則集合と等価な表

現能力を有するが、表現したいω正則集合に対応する時相演算子を導入する必要がある。さらに区間時相論理(ＩＴＬ)[5]の表現能力は真に正則集合を含んでいるが、充足可能性判定問題が決定不能になるという問題点がある。

そこで我々は正則集合と等価な表現能力を持ち、時相演算子３種類のみにより論理式を記述することができる正則時相論理（ＲＴＬ）[6]および、εフリー正則時相論理[7]を提案した。また、その充足可能性判定アルゴリズム[8]、真偽判定アルゴリズムを示し、この真偽判定アルゴリズムを利用した論理設計の検証手法を提案してきた[7]。

しかし、正則時相論理は有限長の系列上で定義されており、無限長の系列上での動作は表現することができない。このため、無限長の系列上の動作と捉えたほうが自然な対象、たとえば順序機械などに対する仕様を記述しようとすると、記述が困難になる場合がある。

そこで本稿では設計対象をω有限オートマトンと捉え、これを記述するために、ＲＴＬを拡張してω正則集合と等価な表現能力を持つω正則時相論理（ωＲＴＬ）を提案する。また、ＲＴＬの場合と同様、ω正則時相論理による形式的検証を行う際にも利用できると考えられる真偽判定アルゴリズムの決定可能性を示す。

以下、２章ではＲＴＬによる形式的検証手法を概観し、その問題点を示す。３章では無限系列に対して拡張したω正則時相論理と、これを用いた順序機械に対する仕様の記述例を示す。４章ではω正則時相論理がω正則集合と等価な表現能力を持つことを示し、さらにその真偽判定問題が決定可能であることを示す。

２．ＲＴＬによる形式的検証とその問題点

本章ではＲＴＬの定義を示し、ＲＴＬにより順序機械に対する仕様を記述した場合の問題点を述べる。なお、以下ではεフリーＲＴＬ[7]を単にＲＴＬと呼ぶ。

2.1 ＲＴＬの定義

本節では、ＲＴＬのシンタクス、及びゼマンティクスを定義する。なお、このシンタクスはωＲＴＬの場合も同一であり、また、ωＲＴＬのセマンティクスはＲＴＬのセマンティクスを用いて定義される。

[定義2.1] ＲＴＬのシンタクス

原始記号は、原始命題ｐ，ｑ，ｒ，…、論理記号～，∨，○，□，：、及び補助記号（，）からなる。ここで、～，∨，○，□，：を各々 not,or,next,repeat,concatenationと呼ぶ。

また、原始命題の集合をＡＰで表現し、以下に定義するＲＴＬ式全体をＬＦと表わす。

$p \in AP$，$\eta, \xi \in LF$のとき、

(1)ｐ，(2)（$\sim\eta$），(3)（$\eta \vee \xi$），(4)（$\bigcirc\eta$），(5)（$\eta : \xi$），(6)（$\Box\eta$）はＬＦの要素であり、以上の(1)－(6)を有限回適用して得られるもののみが、ＲＴＬ式である。 □

また、$\eta \wedge \xi$，$\eta \supset \xi$，$\eta \equiv \xi$，$\forall_T$，$\forall_F$ はそれぞれ通常の意味で and, imply, equivalence,恒真式、恒偽式を表現しているものとする。

単項演算子は２項演算子よりも高い優先順位を持つものとし、論理記号の適用範囲が明確な場合は、補助記号を省略することがある。

つぎに線形時間モデルに基づくＲＴＬの意味を示す。

[定義2.2] ＲＴＬのモデル

Σを状態の有限集合、Ｂ＝｛Ｔ，Ｆ｝を真理値を与える関数$m : \Sigma \times AP \to B$を考える。このとき、$<\Sigma, m>$をＲＴＬのモデルと呼ぶ。 □

モデル$<\Sigma, m>$は各状態$s \in \Sigma$に対する原始命題の真理値を与える。

[定義2.3] ＲＴＬ式の真理値

Σ上の長さ１以上の有限長系列$\sigma \in \Sigma^+$（$\sigma = s_0 s_1 \cdots s_n$とする）に対して、ＲＴＬ式の真理値を与える関数$M_T : \Sigma^+ \times LF \to B$を次のように定義する。

$p \in AP$，$\eta, \xi \in LF$とする。また、$|\sigma|$により、系列σの長さを表わすものとし、$|\sigma| \geq 2$のとき$\sigma_1 = s_1 s_2 \cdots s_n$とする。

(1)$M_T(\sigma, p) = m(s_0, p)$

(2)$M_T(\sigma, \sim\eta) = T$ iff $M_T(\sigma, \eta) = F$

(3)$M_T(\sigma, \eta \vee \xi) = T$ iff

$M_T(\sigma, \eta) = T$ または $M_T(\sigma, \xi) = T$

(4)$M_T(\sigma, \bigcirc\eta) = T$ iff

$|\sigma| \geq 2$ かつ $M_T(\sigma_1, \eta) = T$

(5)$M_\tau(\sigma,\eta:\xi)=T$ iff $M_\tau(\alpha_1,\eta)=T$かつ$M_\tau(\alpha_2,\xi)=T$なる$\alpha_1\in\Sigma^+$および$\alpha_2\in\Sigma^+$が存在して、$\sigma=\alpha_1\alpha_2$と表わせる

(6)$M_\tau(\sigma,\boxplus\eta)=T$ iff

$M_\tau(\alpha_i,\eta)=T$なる$\alpha_i\in\Sigma^+(1\leq i\leq t)$が存在し、$\sigma=\alpha_1\alpha_2\cdots\alpha_t$と表わせる □

$M_\tau(\sigma,\eta)=T$ のとき、$<m,\sigma>\vDash_\tau\eta$と表わし、mが明確な場合は、単に$\sigma\vDash_\tau\eta$と表わす

ＲＴＬ式ηに対し、$<m,\sigma>\vDash_\tau\eta$となるmと$\sigma$が存在するとき、$\eta$は有限系列において充足可能、または単に充足可能であると言うこととする。また、任意のmとσに対して$<m,\sigma>\vDash_\tau\eta$となるとき、$\eta$は有限系列において恒真、または単に恒真であると言い、$\vDash_\tau\eta$と表わす。

[定義2.4]

モデル$<\Sigma,m>$のもとでＲＴＬ式ηの真理値をＴにするようなΣ上の系列全体の集合を$L_\tau<\Sigma,m>(\eta)$とする。即ち、

$L_\tau<\Sigma,m>(\eta)=\{\sigma\mid\sigma\in\Sigma^+,<m,\sigma>\vDash\eta\}$

とし、混乱の恐れがないかぎり$L_\tau(\eta)$と書く。 □

[定義2.5] 表現等価

ＲＴＬが正則集合に表現等価であるとは、任意のm，ηに対して、$L_\tau<\Sigma,m>(\eta)$がΣ上の正則集合であり、かつΣ上の任意の正則集合Ｒに対して、$L_\tau<\Sigma,m>(\eta)=R$となるm，ηが存在することをいう。□

[命題2.1] [6]$L_\tau(\eta)$はΣ上のεフリー正則集合（空系列εを含まない正則集合）である。 □

[定義2.6] [7]構造モデル

ＲＴＬのモデル$<\Sigma,m>$に対して、$S=<\Sigma,m,R,\Sigma_0>$を構造モデルと呼ぶ。ここで、ＲはΣ上の２項関係で、任意の状態$s\in\Sigma$に対して、$(s,s')\in R$となる状態$s'\in\Sigma$が存在するものとする。また、Σ_0はΣの部分集合で初期状態の集合を表わす。 □

[定義2.7] [7]構造モデルＳ上での真理値

構造モデル$S=<\Sigma,m,R,\Sigma_0>$において、状態$s\in\Sigma$からのＳ上の全ての有限長の経路σに対して、$\sigma\vDash_\tau\eta$となるとき、ηは$<S,s>$のもとで真であるという。また、全ての$s_0\in\Sigma_0$に対して、ηが$<S,s_0>$のもとで真になるとき、ηはＳのもとで真であるという。 □

2.2 ＲＴＬによる形式的検証とその問題点

本節では、ＲＴＬの真偽判定を利用した検証手法の概略を示し、検証例として順序機械を取り上げ無限系列上の動作を記述できないことから生じる問題を示す。

以下では、順序機械は完全指定であるとし、Mealy型の状態遷移図の形式で表現されているものとする。

（１）仕様をＲＴＬ式Specで記述する。

この際、状態遷移図における入力と出力の対を{1,0}により適当に符号化し、{1,0}を真理値{T,B}とみなして原始命題を導入する。Specではこの原始命題を用いて記述する。

（２）設計した順序機械の状態遷移図を、対応する可能入出力対グラフ[7]に変換する。

ここで、可能入出力対グラフとは状態遷移図の節点と枝を入れ替えて得られるグラフであり、各節点は状態遷移図の枝をラベル付けしている入出力対によって、ラベル付けされる。また、状態遷移図上の初期状態からの枝に対応する可能入出力対グラフの節点を、初期状態と呼ぶ。

（３）可能入出力対グラフをＲＴＬの構造モデルとみなし、グラフ上で初期状態から始まる有限長の経路全てに対して、Specが真になるかどうかを調べる。つまり、構造モデル上での真偽判定を行う。

以下の例を考える。

[例2.1] 入力Ｘ，出力Ｙとし、原始命題ｘ，ｙの真および偽がそれぞれ１および０に対応しているものとする。また、「常に」の概念を次のように表わす。

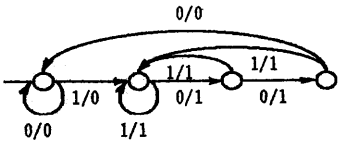
$$\Box\eta \triangleq \sim(\forall_t:\sim\eta)$$

（仕様）

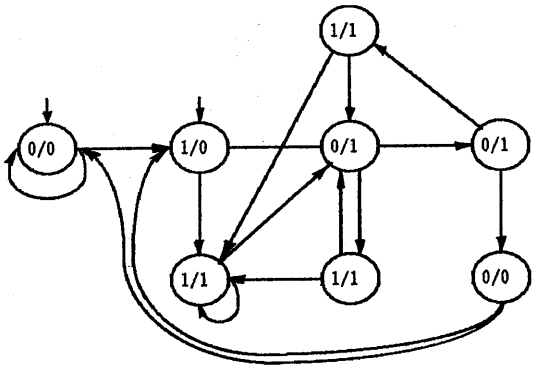
「入力Ｘが１のとき、次の時刻およびその次の時刻の出力Ｙが１である」

これをεフリーＲＴＬ式で表現して、次のように記述したとする。

$$\eta \triangleq (x\supset(\bigcirc y\wedge\bigcirc\bigcirc y))$$



（ａ）状態遷移図



（ｂ）可能入出力対グラフ

図2.1　状態遷移図から状態グラフへの変換

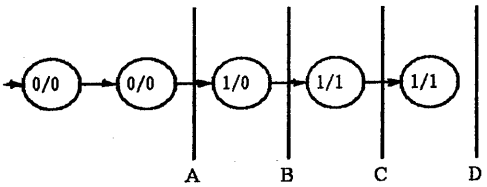


図2.2　□ηが成立しない経路

$$Spec \triangleq \Box\eta \qquad (2.1)$$

これに対して、図2.1(a)の状態遷移図で表わされる順序機械を設計したとする。図2.1(b)がこの順序機械の可能入出力対グラフである。

この可能入出力対グラフを構造モデルとみなして、すべての有限の経路にたいして真偽判定を行うわけであるが、図2.1(b)から図2.2に示す有限の経路を取り出すと、この系列に対して、Specは偽になる。これは、□η が「有限長の経路のどこで分割しても、後半部分でηが真になる」ことを表現しているのに対し、図2.2のＢ－Ｄ間およびＣ－Ｄ間では、系列の長さが短いためηが真にならないからである。Specがこの系列で真になるようにするためにはＢ－Ｄ間およびＣ－Ｄ間では、ηの真偽判定を行わないようにしなければならない。そこで、Specを次のように書かなければならない。

$$Spec \triangleq \Box(\eta \vee len1 \vee \bigcirc len1) \qquad (2.2)$$

（ただし、$len1 \triangleq \sim\bigcirc\forall$ とする）

これらのことは仕様を表現する式は、正しい入出力対の系列上では、有限長の任意の長さに対して真になるようにしなければならず、そのためには順序機械の動作を解析する必要があることを示している。こうした問題が生じるのは、ＲＴＬが有限長の系列のみに対する記述しかできないにもかかわらず、対象が無限時間動作すると捉えて仕様を記述したためである。ＲＴＬを設計検証に用いる場合には、設計対象を有限オートマトンと捉えたが、対象の動作を無限長の系列上に拡張した場合には、ω有限オートマトンと捉えることができる。そこで、以下の章ではω正則集合を表現する能力のあるω正則時相論理について考える。

３．ω正則時相論理

本章では、ω正則時相論理（以下、ωＲＴＬと略記する）と、これを用いた仕様の記述例を示す。

3.1 ωＲＴＬの定義

ωＲＴＬの定義を示す前に、系列の連接、系列集合の連接、閉包、ω閉包を定義する。なお、以下では、Σをアルファベットとし、Σ上の有限長の系列（有限系列と呼ぶ）全ての集合及び無限長の系列（無限系列と呼ぶ）全ての集合をそれぞれΣ^+とΣ^ωにより表わし、$\Sigma^\infty = \Sigma^+ \cup \Sigma^\omega$とする。このとき、$|\sigma|$は系列$\sigma \in \Sigma^\infty$の長さを表すものとし、σが無限系列の場合は$|\sigma| = \omega$と表記する。

［定義3.1］α，$\beta \in \Sigma^\infty$、Ｕ，$V \subseteq \Sigma^\infty$とし、α(i)は系列のｉ番めの文字を意味するものとする。

・$|\alpha| = \omega$　ならば　$\alpha\beta = \alpha$

そうでなければ　$\alpha\beta(i) = \alpha(i)$ for $0 < i < |\alpha|$

かつ $\alpha\beta(|\alpha|+i)=\beta(i)$ for $0<i<|\beta|$

・$UV=\{uv \mid u\in U$ かつ $v\in V\}$

・$U^{+}=\{u_1$を有限個連接したもの，$u_1\in U\}$

・$U^{\omega}=\{u_1u_2\cdots \mid u_j\in U\cap\Sigma^{+}$ for all $j\}$ □

ωＲＴＬのシンタクスとモデルの定義はそれぞれ、定義2.1，2.2と同一である。

［定義3.2］ωＲＴＬ式の真理値

Σ上の無限系列$\sigma\in\Sigma^{\omega}$（$\sigma=s_0s_1s_2\cdots$と記す）に対し、ωＲＴＬ式の真理値を与える関数$M_\omega:\Sigma^{\omega}\times LF\rightarrow B$を次のように定義する。以下で用いる関数$M_f:\Sigma^{*}\times LF\rightarrow B$は定義2.3で示したものである。

$p\in AP$，$\eta,\xi\in LF$とする。また、$\sigma_1=s_1s_2\cdots$とする。

(1)$M_\omega(\sigma,p)=m(s_0,p)$

(2)$M_\omega(\sigma,\sim\eta)=T$ iff $M_\omega(\sigma,\eta)=F$

(3)$M_\omega(\sigma,\eta\vee\xi)=T$ iff

$M_\omega(\sigma,\eta)=T$ または $M_\omega(\sigma,\xi)=T$

(4)$M_\omega(\sigma,\bigcirc\eta)=T$ iff

$|\sigma|\geqq 2$ かつ $M_\omega(\sigma_1,\eta)=T$

(5)$M_\omega(\sigma,\eta:\xi)=T$ iff

$M_f(\alpha_1,\eta)=T$かつ$M_\omega(\alpha_2,\xi)=T$なる$\alpha_1\in\Sigma^{+}$および$\alpha_2\in\Sigma^{\omega}$が存在して、$\sigma=\alpha_1\alpha_2$と表わせるか、

または

$|\sigma|=\omega$ かつ $M_\omega(\sigma,\eta)=T$が成立する。

(6)$M_\omega(\sigma,\boxplus\eta)=T$ iff

$M_f(\alpha_i,\eta)=T$なる$\alpha_i\in\Sigma^{+}(1\leqq i\leqq t-1)$ 及び$M_\omega(\alpha_t,\eta)=T$なる$\alpha_t\in\Sigma^{\omega}$が存在し、$\sigma=\alpha_1\alpha_2\cdots\alpha_t$と表わせるか

または

$|\sigma|=\omega$ かつ $M_f(\alpha_i,\eta)=T$なる$\alpha_i\in\Sigma^{+}$ $(i=1,2,\cdots)$が無限個存在し$\sigma=\alpha_1\alpha_2\cdots$と表わせる□

$M_\omega(\sigma,\eta)=T$ のとき、$<m,\sigma>\vDash_\omega\eta$と表わし、mが明確な場合は、単に$\sigma\vDash_\omega\eta$と表わす。また、無限系列に対する充足可能性、恒真性もＲＴＬの場合と同様に定義する。

図3.1にωＲＴＬのセマンティクスを直観的に表わしたものを示す。$\sigma\vDash_\omega\bigcirc\eta$は２つ目以降の状態系列で$\eta$が真であることを、$\sigma\vDash_\omega\eta:\xi$は$\eta$を真にする有限系列と$\xi$を真にする無限系列の連接であるか、または$\eta$を真にする無限系列であることを意味する。$\sigma\vDash_\omega\boxplus\eta$は$\eta$を真にする有限系列（０個以上で有限個）と$\eta$を真にする無限系列の連接であるか、または$\eta$を真にする有限系列無限個の連接であることを意味する。

ωＲＴＬを用いて検証を行う場合にも、2.2節で示したＲＴＬの場合と同様、構造モデル上での真偽判定アルゴリズムを用いることが考えられる。そこでＲＴＬと同一の構造モデル（定義2.6）を考え、このモデル上での真理値を定義しておく。

［定義3.3］構造モデルＳ上での真理値

構造モデル$S=<\Sigma,m,R,\Sigma_0>$において、状態$s\in\Sigma$からのＳ上の全ての無限長の経路σに対して、$\sigma\vDash_\omega\eta$となるとき、ηは$<S,s>$のもとで真、全ての$s_0\in\Sigma_0$に対して、ηが$<S,s_0>$のもとで真になるとき、ηはＳのもとで真であるという。 □

構造モデル$S=<\Sigma,m,R,\Sigma_0>$とηが与えられたとき、Ｓ上での真偽判定をω**真偽判定**と呼ぶことにする。

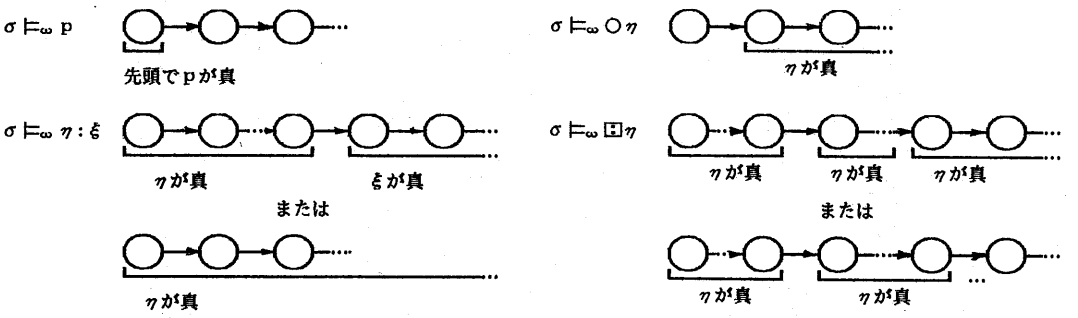


図3.1 ωＲＴＬの直観的な意味

3.2 ωＲＴＬによる記述

系列の長さが１であるという性質len1は次のように定義することができる。

$$len1 \triangleq \sim \bigcirc \forall_\eta \qquad (3.1)$$

また、系列の長さが無限または有限であるという性質InfiniteおよびFiniteは

$$Infinite \triangleq (\forall_\eta : \forall_\eta) \qquad (3.2)$$

$$Finite \triangleq \sim Infinite \qquad (3.3)$$

このとき、

$$\vdash_\omega Infinite \equiv \forall_\eta$$

$$\vdash_\omega Finite \equiv \forall_\eta$$

これらを利用すると、従来の時相論理の□（常に）は次のように定義できる。

$$\Box\eta \triangleq \eta \wedge \sim (Finite : \sim\eta) \qquad (3.4)$$

また、性質ηを持つ有限長の区間が無限回存在するという性質は、

$$\boxplus(\eta \wedge Finite)$$

と表現できる。このようにωＲＴＬは無限の系列に対する性質を表現することができる。

例2.1について、ωＲＴＬを利用して検証することを考えると、これは可能入出力対グラフを構造モデルとみなし、Specに対してω真偽判定を行うことである。この場合、構造モデル上の無限長の系列のみを考えればよいので、式(2.1)をSpecとしてそのまま、用いることができる（ただし□の定義は式(3.4)による）。

次に順序機械に対する仕様記述例を示す。

［例3.1］　次の仕様を満足する順序機械を考える。

（言葉による仕様）　入力はＸ、出力はＹ、初期値はＸ＝Ｙ＝０とする。１が３回入力されたら、３回目の入力と同時に１を出力する。

（ωＲＴＬによる仕様）　入力Ｘ、出力Ｙに対して、原始命題ｘ，ｙを導入し、Ｘ，Ｙが１のとき真、０のとき偽になるとする。

$$\eta \ until \ \xi \triangleq \xi \vee (\Box\eta : \xi)$$

$$Pulse\_x \triangleq (\sim x \ until \ (x \wedge len1))$$

仕様を表現する式ををSpecとすると、

$$Spec \triangleq \boxplus Pulse\_x : Pulse\_x \ : \ Pulse\_x \ \Box \sim y \quad (x \equiv y)$$

□

4　ω正則時相論理の諸性質

本章ではω正則集合とωＲＴＬの関係を述べ、ωＲＴＬの真偽判定問題が決定可能であることを示す。

4.1 ω正則集合とωＲＴＬの関係

ω正則集合を次のように定義する。

［定義4.1］［9］ω正則集合

（Σ上のω正則集合のクラス）

＝｛$B_i C_i^\omega$の有限和｜B_i, C_iは（Σ^+上の）εフリー正則集合｝　□

また、ω有限オートマトンを次のように定義する。

［定義4.2］［9］ω有限オートマトン

Ａ＝（Ｑ，Σ，δ，q_0，Ｆ）を有限オートマトンとする。ここでＱとΣはそれぞれ状態と入力記号の有限集合、δ：Ｑ×Σ→2^Qは状態遷移関数、q_0は初期状態、Ｆは受理状態の集合を示す。また、$x \in \Sigma^\omega$に対してｘを初期状態から入力するとき、無限回通過する状態の集合をIn(x)と表現する。この時Ａが受理する言語｜Ａ｜は次のように定義される。

$$|A| = \{x \in \Sigma^\omega \mid In(x) \cap F \neq \phi\} \qquad \Box$$

［命題4.1］［9］　次の(１)と(２)は等価である。

(１)Ｕがω正則集合である。

(２)あるω有限オートマトンが存在し、Ｕを受理する

また、Ｕからω有限オートマトンを構成するアルゴリズムおよびω有限オートマトンからＵを構成するアルゴリズムが存在する。　□

［命題4.2］［9］　ω正則集合は集合のブール演算に関して閉じている。　□

［定義4.3］

$L_\omega < \Sigma, m > (\eta) = \{\sigma \mid \sigma \in \Sigma^\omega, < m, \sigma > \vdash \eta\}$

とし混乱の恐れがないかぎり、$L_\omega(\eta)$と書く。　□

［定理4.1］$L_\omega(\eta)$はΣ上のω正則集合である。

（証明）

(1)$p \in AP, S(p) = \{s \mid s \in \Sigma, m(s,p) = T\}$とすると、$L_\omega(p) = S(p)\Sigma^\omega$

(2)$L_\omega(\sim\eta) = \Sigma^\omega - L_\omega(\eta)$

(3) $L_\omega(\eta \vee \xi) = L_\omega(\eta) \cup L_\omega(\xi)$

(4) $L_\omega(\bigcirc\eta) = \Sigma L_\omega(\eta)$

(5) $L_\omega(\eta : \xi) = L_r(\eta) L_\omega(\xi) \cup L_\omega(\eta)$

(6) $L_\omega(\boxplus\eta) = L_r(\eta)^+ L_\omega(\eta) \cup L_\omega(\eta)$
$\cup L_r(\eta)^\omega$

このとき、(1)がω正則集合であるのはあきらか。$L_\omega(\eta)$及び$L_\omega(\xi)$がω正則集合であれば(3)－(6)の左辺もまたω正則集合である。さらに命題4.2により、(2)の左辺もω正則集合である。 □

また定理4.1の逆の定理も成立する。

[定理4.2] Σ上の任意のω正則集合Rに対して、あるモデル$<\Sigma, m>$が存在して、$<m, \sigma> \models_\omega \xi(R)$ iff $\sigma \in R$ なるωＲＴＬ式$\xi(R)$が存在する。

（証明）全てのｓ∈Σに対して、次のような原始命題を導入する。

$m(s', p_s) = T$ iff $s' = s$

また、このようなp_sの集合をＡＰ。と書くことにする。

定義4.1より、$R = R'R''^\omega$（ただし、$R', R'' \subseteq \Sigma^+$は正則集合）と仮定してよい。まず、R',R"に対して$\xi(R'), \xi(R'')$を次の手続きにより帰納的に構成する。以下では、$s \in \Sigma$、R_1, $R_2 \subseteq \Sigma^+$は Σ上の正則集合とする。

（１）$\xi(\phi) = \forall_r$

（２）$\xi(s) = p_s \wedge (\sim \bigcirc \forall_r)$ ただし、$p_s \in AP_s$

（３）$\xi(R_1 R_2) = \xi(R_1) : \xi(R_2)$

（４）$\xi(R_1 + R_2) = \xi(R_1) \vee \xi(R_2)$

（５）$\xi(R_1^+) = \boxplus\xi(R_1) \wedge$ Finite

このとき、求めるωＲＴＬ式$\xi(R)$は次のように定められる。

$$\xi(R) \triangleq ((\xi(R'') \equiv \forall_r) \supset \forall_r)$$
$$\wedge$$
$$((\xi(R'') \not\equiv \forall_r) \supset (\xi(R') : \boxplus\xi(R'')))$$

□

ωＲＴＬとω正則集合における等価性を定義2.5と同様に定義すると次の系を示すことができる。

[系4.1] ωＲＴＬはω正則集合と表現等価である。□

4.2 真偽判定問題について

本節では、まずωＲＴＬの充足可能性判定アルゴリズムが決定可能であることを示す。さらに、構造モデル上での真偽判定問題の決定可能性を、充足可能性判定問題に帰着することにより証明する。

[補題4.1] あるω有限オートマトン$A = (Q, \Sigma, \delta, q_0, F)$に対して、次のような条件$P(t)$を考える。Qの状態数をｎ、ｔ∈Ｆとする。

$P(t)$＝「有限オートマトン$B = (Q, \Sigma, \delta, q_0, \{t\})$ または $C = (Q, \Sigma, \delta, t, \{t\})$が長さn-1以下のどんな系列も受理しない」

このとき、次の(1)と(2)は同値である。

(1) $\forall t \in F : P(t)$が成立する

(2)ω有限オートマトンＡが受理する言語が空。

（証明）(1)→(2) 任意の（ωでない）有限オートマトン（状態数ｎ）は、長さｎ－１以下のどんな系列も受理しないならば、それが受理する言語は空である[10]。これより、条件(1)が成立しているならば、どのｔ∈Ｆに対しても、q_0からｔへの状態遷移を引き起こす入力系列が存在しないか、あるいはｔを無限回通過するような入力系列は存在しない。従って(2)が成立する。

(2)→(1) ∃ｔ∈Ｆ：有限オートマトン$B = (Q, \Sigma, \delta, q_0, \{t\})$と$C = (Q, \Sigma, \delta, t, \{t\})$が、それぞれ長さｎ－１以下の系列$\beta$と$\gamma$を受理するとすると、ω有限オートマトンＡは$\beta\gamma^\omega$を受理する。 □

[定理4.3] ω充足可能性判定問題は決定可能である。

（証明）ωＲＴＬ式ηに対して$L_\omega(\eta)$を構成する。命題4.1より、これを受理するω有限オートマトンＡを構成することができる。Ａに対して補題3.2の条件(1)を考えると、(1)が成立するかどうかは明らかに決定可能である。したがって、$L_\omega(\eta)$の空判定問題、すなわちω充足可能性判定問題は決定可能である。□

次に、真偽判定について述べる。

[定理4.4] 構造モデル上でのω真偽判定問題は、ω充足可能性判定問題に帰着することができる。したがって、また決定可能である。

（証明）構造モデル$S = <\Sigma, m, R, \Sigma_0>$において、Ｓの状態ｓに１対１に対応する原始命題$p_s$を導入す

る。ただし、$AP_s = \{p_s \mid s \in \Sigma\}$ として、$AP_s \cap AP = \phi$（空集合）とする。

$$f_x \triangleq \bigwedge_{\{q \mid m(x,q)=T\}} q \quad \wedge \bigwedge_{\{q \mid m(x,q)=F\}} (\sim q)$$

$$g_x \triangleq \bigvee_{(x,y)\in R} p_y$$

$$\eta_x \triangleq \Box\{p_x \supset (f_x \wedge \bigcirc g_x)\}$$

$$\eta_d \triangleq \Box[\bigwedge_{x\in\Sigma}\{p_x \supset (\bigwedge_{y\neq x} \sim p_y)\}]$$

$$\eta_1 \triangleq (\bigvee_{y\in\Sigma} p_y) \wedge (\bigwedge_{x\in\Sigma} \eta_x) \wedge \eta_d$$

$$\eta_2 \triangleq \eta_1 \wedge \eta \wedge \{\bigvee_{s \mid s\in\Sigma_0} p_s\}$$

このとき、ηがS上で真になることとη_2が充足可能であることが等価になる。 □

5．むすび

我々は設計対象を有限オートマトンと捉えて、形式的検証を行うため、正則集合と等価な表現能力を持つＲＴＬを提案し、またこれを用いた検証手法を示してきた。しかし、ＲＴＬは有限長の系列のみしか表現できないため、無限長の系列上での動作であると捉えたほうが自然であるような対象を記述することが、困難なことがある。そこで、本稿では設計対象をωオートマトンと捉え、ＲＴＬを拡張してω正則集合を表現することができるω正則時相論理（ωＲＴＬ）を提案した。また、形式的検証を行う際に利用することができる真偽判定問題が決定可能であることを示した。

本稿では、真偽判定問題を充足可能性判定問題の帰着しているが、真偽判定問題を論理設計検証の場面に応用するためには、より直接的に真偽判定を行って、計算量を削減する必要があり、このアルゴリズムの作成が今後の課題となっている。また、ωＲＴＬは有限長の系列を表現することができないため、有限および無限系列上の動作の記述および形式的検証に際しては、ＲＴＬとωＲＴＬを組み合わせた手法も考えられる。

さらに、正則時相論理による一般的な仕様記述手法も、考察してゆくべき課題となっている。

謝辞　種々御議論頂いた高木直史博士、石浦菜岐佐氏を始め、本学矢島研究室の皆様に深謝致します。

参考文献

[1]N.Rescher and A.Urquhart:Temporal Logic, Springer-Verlag,1971

[2]D.L.Dill and E.M.Clarke：Automatic Verification of Asynchronous Circuits Using Temporal Logic, IEE Proceedings,vol.133,No.5,pp276-282,1986.

[3]M.Fujita et.al.：Aid to Hierarchial and Structured Logic Design Using Temporal Logic and Prolog, IEE Proceedings,vol.133,No.5,pp.283-294, 1986.

[4]P.Wolper:Temporal Logic can be more expressive, Proc, 22nd Annual Symp.on Foundation of Computer Science, pp.340-348,1981.

[5]B.Moszkowski:Reasoning about Digital Circuits, STAN-CS-83-970.1983

[6]平石、矢島：正則集合と表現等価な正則時相論理ＲＴＬ，情報処理，28,2,pp.117-123.

[7]平石、矢島：正則時相論理による論理設計検証について，comp87-67,1987.

[8]平石、濱口、矢島：正則時相論理の充足可能性判定アルゴリズム，COMP87-2, 1987.

[9]S.Eilenberg：Automata, Languages and Machines Vol.A, Academic Press,1974.

[10]A.V.Aho, J.E.Hopcroft and J.D.Ullman：The Design and Analysis of Computer Algorithms,Addison-Wesely,1974.