リアクティブ時相論理

内平直志[1] 本位田真一[2]

情報処理振興事業協会 (IPA)
新ソフトウェア構造化モデル研究本部

リアクティブ・システムの本質は，システム内に観測不可能あるいは制御不可能な要素を含む点である．本報告では，観測可能性（情報の局所性）に注目し，観測可能性／不可能性を明示的に記述できるように分枝時間時相論理を拡張したリアクティブ時相論理を提案する．まず，並行システムのモデルであるラベル付き遷移システムを，ゲーム理論の「情報構造」の概念を用いて抽象化する．次に，情報構造付き遷移システムをモデルとするリアクティブ時相論理を提案する．分枝時間時相論理のモデル検査法と同じ原理で，情報構造付き遷移システムΓがリアクティブ時相論理式 fのモデルであるか ($\Gamma \models f$) を判定することが可能である．また，リアクティブ時相論理を用いたリアクティブ・システムの制御問題の定式化の例を示す．

Reactive Temporal Logic

Naoshi Uchihira Shinichi Honiden

Laboratory for New Software Architectures
Information-technology Promotion Agency (IPA)

Shuwa-Shibakoen 3-chome Bldg., 3-1-38 Shibakoen, Minato-ku, Tokyo 105, JAPAN
E-mail: uchi@caa.ipa.go.jp

This paper proposes a new temporal logic for specifying reactive systems, called Reactive Temporal Logic (RTL). RTL features partiality of information such as unobservability and uncertainty, which are the nature of reactive systems. First, we formalize partiality of information by introducing information structure to a transition system. Then, RTL is defined, whose model is given as a transition system with information structure. As RTL relies on the branching time temporal logic, we can apply the model checking method to verify that a transition system with information structure satisfies a RTL formula. We show how to apply RTL to formalization of control problem of reactive systems.

[1]株式会社 東芝 から出向
[2]株式会社 東芝 から出向

1 はじめに

リアクティブ・システム (Reactive System) とは，システムの外界 (Environment) と継続的な相互作用を行なうシステムであり，年々そのニーズが高まっている．例えば，計算機をネットワークで接続する分散型システムにおいて，各計算機は，自分が接続されているネットワークを外界とするリアクティブ・システムである．また，シーケンス制御システムは，制御対象を外界とする典型的なリアクティブ・システムである．従来，このようなシステムを，一括してリアルタイム・システムと称してきたが，ここでは，非決定性な割り込みなどの相互作用が本質的なシステムをリアクティブ・システムと呼び，その中でも特に定量的な時間が本質的なシステムをリアルタイム・システムと呼ぶ．リアクティブ・システムの特徴は，システム内に観測不可能または制御不可能な要素を含む点である．すなわち，リアクティブ・システムにおいて，外界の情報を即時的かつ完全に観測することは不可能であり，また，外界が主導的に起動するアクションは制御不可能である．この観測不可能性（情報の局所性）と制御不可能性がリアクティブ・システムと事務計算などの非リアクティブ・システムとの違いの本質であり，設計／プログラミングを難しくしている点である．

一般に，リアクティブ・システムはいかなる外界の変化にも安全に対応できるための高信頼性が要求されている．高信頼化のための１つのアプローチとして，リアクティブ・システムの仕様を形式的に記述し，システムの検証や合成を行なう手法が有望であると思われる．

システムの動的挙動に関する仕様や制約を形式的に記述する有望な手法の１つに時相論理 (Temporal Logic) がある．時相論理は，１９７０年後半から，計算機科学への適用が活発に研究され現在に至っている．具体的な成果としては，逐次／並行プログラム，通信プロトコル，論理回路などの時間的挙動が本質であるような動的システムの仕様記述および合成，検証がある（[1] [2] [3] などの文献を参照されたし）．最近，時相論理の拡張として，定量的な時間が記述できるリアルタイム時相論理がいくつか提案されている [4][5]．しかし，リアクティブ・システムの本質である観測可能性や制御可能性を明示的に扱える時間論理はなかった．

本報告の目的は，リアクティブ・システムを形式的に記述し，検証や合成を可能にするための新しい時相論理（リアクティブ時相論理: Reactive Temporal Logic）を提案することである．

2 情報集合を持つ遷移システム

並行システムのモデルとして用いられるラベル付き遷移システム (Labeled Transition System: LTS) を，観測可能性の観点から抽象化する．すなわち，ラベルの代わりにゲーム理論における情報構造 (Information Structure: IS) を導入し，情報集合を持つ遷移システムを定義する．リアクティブ時相論理は，この情報構造を持つ遷移システムをモデルとするように拡張された分枝時間時相論理 (Branching Time Temporal Logic) である．

2.1 ゲーム理論における情報構造

ゲーム理論の例として有名な「囚人のジレンマ」を用いて，ゲーム理論における「情報構造」を説明する．このゲームは，逮捕された共犯の２人の囚人が独立に取り調べをうけている状況をモデル化したものである．各囚人は,「自白」するか「黙秘」するかのいづれかを選択しなければならない．このときのゲームの利得行列は (表 1) のようになる．このとき，両者が黙秘を貫けば全体としての最適解 (-1,-1) であるにもかかわず，相手の選択を知ることができない（観測不可能である）ため，相手の裏切りを恐れて両者自白する局所最適（パレート最適）解 (-5,-5) に陥ってしまう，という現象がこのゲームの特徴である．このゲームは図１のような遷移システムと情報構造で表現できる．ここで，遷移システムのいくつかの状態を線で囲んだものが情報構造であり，１つの線で囲まれたいくつかの状態は観測者にとって識別不可能である．例えば，囚人２にとっては囚人１の選択（自白したか黙秘したか）が観測不可能であるため，状態 s_1 と状態 s_2 は１つの線で囲まれる．ゲーム理論では，この遷移システムと情報構造を用いたゲームの表現方法を展開形ゲームと呼ぶ [8]．

表 1: 囚人のジレンマ

		囚人 2 (P2)	
	選択	黙秘 *(m)*	自白 *(c)*
囚人 1 (P1)	黙秘 (m)	(-1,-1)	(-10,0)
	自白 (c)	(0,-10)	(-5,-5)

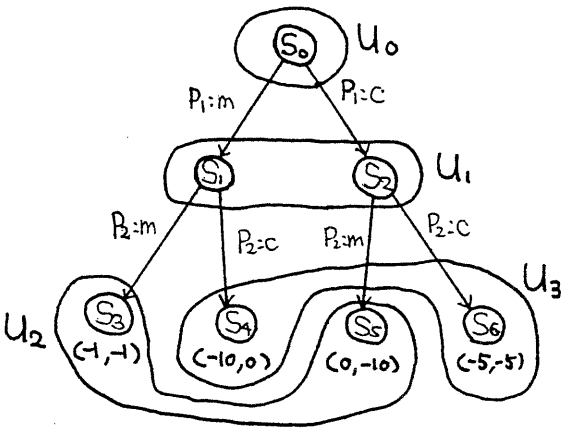


図 1: 囚人のジレンマの展開形ゲームによる表現

2.2 情報構造付き遷移システム

囚人のジレンマで例示した情報構造を持つ遷移システム (Transition System with Information Structure : TSIS) を定式化する．さらに，情報構造を持つ遷移システムが，ラベル付き遷移システムの観測可能性の観点からの抽象化になっていることを示す．さらに，ラベル付き遷移システムのラベルの観測という立場からは表現できなかった「現在の非決定性」が表せるようになったことを示す．

定義 1 (遷移システム)
$T = (S, \delta)$: 遷移システム,
S : 状態集合,
$\delta \subset S \times S$: 状態遷移を表す 2 項関係.
ここで，$(s, s') \in \delta$は，s から s'への状態遷移を表す．また，$(s, s') \in \delta$と記す代わりに $s \rightarrow s'$と記すこともできる．

遷移システム T の情報構造は，状態集合 S の分割として表される．ここで，異なる区分に属する 2 つの状態は観測上識別可能だが，同じ分割区分に属する 2 つの状態は観測者にとって識別不可能である．すなわち，区分は観測上 1 つの状態とみなすことができる．以下では，各区分を観測状態 (Observational State) と呼び，実際の遷移システムの状態を実状態 (Real State) と呼ぶことにする．

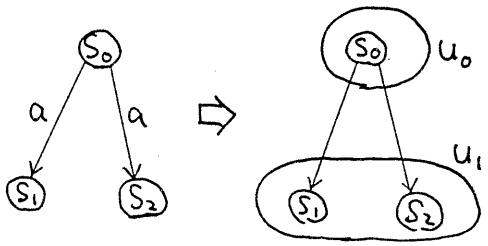
定義 2 (情報構造付き遷移システム)
$\Gamma = (T, U, \Phi, u_0)$,
$T = (S, \delta)$: 遷移システム,
U : 観測状態集合,
$\Phi : S \rightarrow U$ 観測関数,
$u_0 \in U$: 初期観測状態.

例 1 (囚人のジレンマの TSIS による表現)
$\Gamma = (T, \{u_0, u_1, u_2, u_3\}, \Phi, u_0)$,
$T = (\{s_0, s_1, s_2, s_3, s_4, s_5, s_6\}, \delta)$,
$\delta : s_0 \rightarrow s_1, s_0 \rightarrow s_2, s_1 \rightarrow s_3, s_1 \rightarrow s_4, s_2 \rightarrow s_5,$
$s_2 \rightarrow s_6, \Phi(s_0) = u_0, \Phi(s_1) = u_1, \Phi(s_2) = u_1,$
$\Phi(s_3) = u_2, \Phi(s_4) = u_3, \Phi(s_5) = u_2, \Phi(s_6) = u_3.$

2.3 情報構造付き遷移システムの特徴

2.3.1 ラベル付き遷移システムの抽象化

情報構造を持つ遷移システムの特徴の 1 つは，ラベル付き遷移システムの観測の観点からの抽象化になっている点である．従来，ラベル付き遷移システムの観測上の等価性に関しては，トレース等価 (Trace Equivalence)，双模倣等価 (Bisimulation Equivalence) や試験等価 (Testing Equivalence) などとして定式化されている [6]．これらの定式化では，外界から観測できる動作（ラベル名で認識される）系列からある判断基準で識別できない状態は等価な状態とみなす．これらの等価性判定では，ラベル名が大きな意味を持っている．例えば，図 2(a) のラベル付き遷移システムにおいて s_0から動作 a で遷移可能な状態 s_1と状態 s_2は，動作ラベル a だけしか観測できない観測者からはどちらに遷移したかを識別できない．このとき，ラベル名 a は本質的でなく，状態 s_1と s_2が観測者からは識別できない点が本質である．そこで，ラベル名抽象化し，図 2 (b) のように観測上識別できない状態の集合を観測状態 u_1としたものが，情報構造付き遷移システムである．



(a)LTS (b)TSIS

図 2: LTS と TSIS

ここで，注意すべき点は，観測状態に属する「識別できない状態」と双模倣等価などによる「等価な状態」の違いである．前者は，現在までの観測で識別できない状態であり，後者は，現在以降の観測で識別できない状態である．

また，図3に示す情報構造は，一旦識別できた状態から識別できない状態への遷移を意味する．これは，過去の観測状態の履歴を忘れてしまったことを意味する．多くの場合，過去の観測状態は全て記憶することができると仮定するのが自然である．そこで，以下の完全記憶条件を定義する．

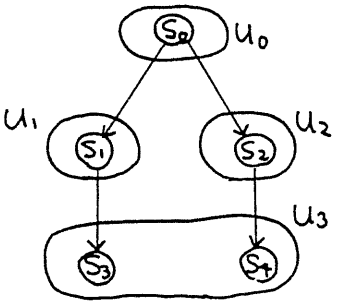


図 3: 不完全な記憶を持つ TSIS

定義 3 (完全記憶条件)
$\Gamma = (T, U, \Phi, u_0)$ において $\forall u \in U.\forall s_1, s_2 \in \Phi^{-1}(u).$ $\forall s_1'.\forall s_2'.($ *if* $s_1' \rightarrow s_1 \wedge s_2' \rightarrow s_2$ *then* $\Phi(s_1') = \Phi(s_2'))$ が成り立つならば，Γは完全記憶条件を満たす．

図 2(a) に示すラベル付き遷移システムは，図 2(b) に示す完全記憶条件を満たす情報構造を持つ遷移システムに抽象化された．一般に，ラベル付き遷移システム Tから，そのラベル意味する本質的情報を保存し完全記憶条件を満たす情報構造付き遷移システム$\Gamma(T)$を構成できる．ここで，遷移システムの１つの実状態が，複数の観測状態に含まれる場合は，その観測状態の数だけ実状態の複製を生成するので，$\Gamma(T)$ は Tより実状態数が増えることが多い（図 4）．

2.3.2 未来の非決定性と現在の非決定性

もう１つの情報構造付き遷移システムの特徴は，現在の非決定性の表現である．前出の図 2(a) のラベル付き遷移システムでは，動作 a によって遷移する先の状態は非決定的に s1 か s2 に定まる．これは，ラベル付き遷移システムでも，未来の非決定性は表現できたことを意味する．しかし，現在の非決定性は表現できなかった．しかし，TSIS では図 5に示すように，「現在，s_1にいるのか s_2にいるのかが，観測者にとって識別できない」が表現可能である．

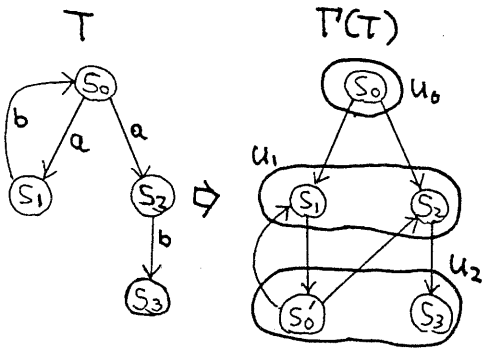


図 4: Tから$\Gamma(T)$ へ抽象化

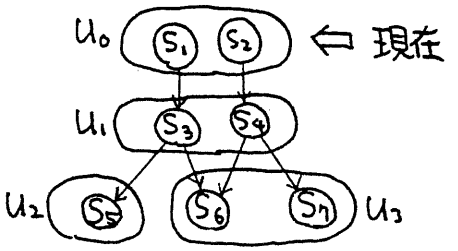


図 5: 現在の非決定性

2.4 情報構造付き遷移システムの等価性

情報構造を持つ遷移システムの双模倣等価性を定義する．これは，ラベル付き遷移システムの双模倣等価性の抽象化になっている．

定義 4 (観測状態双模倣)
二項関係 $R_u \subset U \times U$ が $\Gamma = (T, U, \Phi, u_0)$ における観測状態双模倣であるとは，$R_s \subset S \times S$ が存在し，以下の条件を満たすことである：

- $\forall u_1, u_2 \in U$に対して，
  $u_1 R_u u_2 \Leftrightarrow$
  $\forall s_1 \in \Phi^{-1}(u_1).\exists s_2 \in \Phi^{-1}(u_2).s_1 R_s s_2$, *and*
  $\forall s_2 \in \Phi^{-1}(u_2).\exists s_1 \in \Phi^{-1}(u_1).s_1 R_s s_2$.

- $\forall s_1, s_2 \in S$に対して，
  $s_1 R_s s_2 \Leftrightarrow$
  $\Phi(s_1) R_u \Phi(s_2)$, *and*

$\forall s'_1 \in S.\exists s'_2 \in S.$
(if $s_1 \rightarrow s'_1$ then $s_2 \rightarrow s'_2 \wedge s'_1 R_s s'_2$), *and*
$\forall s'_2 \in S.\exists s'_1 \in S.$
(if $s_2 \rightarrow s'_2$ then $s_1 \rightarrow s'_1 \wedge s'_1 R_s s'_2$).

定義 5 (観測状態双模倣等価性)
観測状態 u_1 と u_2 が観測状態双模倣等価 *($u_1 \approx_o u_2$)* であるとは，$u_1 R_u u_2$ なる観測状態双模倣 $R_u \in U \times U$ が存在することである．
また，$\Gamma_1 = (T_1, U_1, \Phi_1, u_{01})$ と $\Gamma_2 = (T_2, U_2, \Phi_2, u_{02})$ に対して，$u_{01} \approx_o u_{02}$ ならば $\Gamma_1 \approx_o \Gamma_2$ とする．

例 2 (観測状態双模倣等価性)
以下の Γ' は因人のジレンマ Γ と観測状態双模倣等価な *TSIS* である．*(すなわち，$\Gamma \approx_o \Gamma', u_0 \approx_o u'_0, u_1 \approx_o u'_1, u_2 \approx_o u'_2, u_3 \approx_o u'_2$)*:
$\Gamma' = (T', \{u'_0, u'_1, u'_2\}, \Phi', u'_0)$,
$T' = (\{s'_0, s'_1, s'_2\}, \delta')$,
$\delta' : s'_0 \rightarrow s'_1, s'_1 \rightarrow s'_2$,
$\Phi'(s'_0) = u'_0, \Phi'(s'_1) = u'_1, \Phi'(s'_2) = u'_2$.

例 2では，Γ と Γ' は，2回の遷移でデッドロックになるという意味で等価である．もちろん，利得を表す状態属性を考慮すると等価ではなくなるかもしれない．このような，状態属性を持つ遷移システムに関しては後に述べる．

命題 1 (LTS と TSIS の等価性の関係)
ラベル付き遷移システム $T1$ と $T2$ に対して，
$T1 \approx T2 \Leftrightarrow \Gamma(T1) \approx_o \Gamma(T2)$ である．
(ここで，$\approx$ はラベル付き遷移システムの双模倣等価)

3 リアクティブ時相論理

3.1 リアクティブ時相論理

情報構造付き遷移システムをモデルとする分枝時間時相論理を提案する．これをリアクティブ時相論理 (RTL) と呼び，構文規則と意味を分枝時間時相論理 CTL の定義 [7] に準拠して定義する．

3.1.1 構文規則

従来の CTL などの分枝時間時相論理では，論理式は状態論理式とパス論理式から構成されていたが，リアクティブ時相論理式では新たに「観測状態式」を導入した点が特徴である．

観測状態式
- $true$ は観測状態式．
- f_1 と f_2 が観測状態式ならば，$f_1 \wedge f_2$ と $\neg f_1$ も観測状態式．
- h が実状態式ならば，$\exists_o h$ は観測状態式．

実状態式
- $true$ は実状態式．
- h_1 と h_2 が実状態式ならば，$h_1 \wedge h_2$ と $\neg h_1$ は実状態式．
- h が実状態式ならば，$\exists X h$ は実状態式．
- f が観測状態式ならば，$\exists X_o f$ は実状態式．

RTL 式
- f が観測状態式ならば，f は RTL 式．

マクロ表現
- $false = \neg true$
- $f_1 \vee f_2 = \neg(\neg f_1 \wedge \neg f_2)$
- $f_1 \supset f_2 = \neg f_1 \vee f_2$
- $\forall f = \neg\exists\neg f$
- $\forall_o f = \neg\exists_o\neg f$

3.1.2 意味

定義 6 ($[\![f]\!]_U$)
各 *RTL* 式 f に対して，その式の意味 $[\![f]\!]_U \subset U$ (Γ において式 f が成り立つ観測状態の集合) は下記のように定義される．ここで，$[\![h]\!]_S$ は Γ において実状態式 h が成り立つ実状態の集合とする．

$[\![true]\!]_U = U$
$[\![\neg f]\!]_U = U - [\![f]\!]_U$
$[\![f_1 \wedge f_2]\!]_U = [\![f_1]\!]_U \cap [\![f_2]\!]_U$
$[\![\exists_o h]\!]_U = \{u \in U \mid \exists s \in \Phi^{-1}(u).s \in [\![h]\!]_S\}$
$[\![true]\!]_S = S$
$[\![\neg h]\!]_S = S - [\![h]\!]_S$
$[\![h_1 \wedge h_2]\!]_S = [\![h_1]\!]_S \cap [\![h_2]\!]_S$
$[\![\exists X h]\!]_S = \{s \in S \mid \exists s' \in S.s \rightarrow s' \wedge s' \in [\![h]\!]_S\}$
$[\![\exists X_o f]\!]_S = \{s \in S \mid \exists s' \in S.s \rightarrow s' \wedge \Phi(s') \in [\![f]\!]_U\}$

定義 7 (モデル)
$\Gamma = (T, U, \Phi, u_0)$ と $f \in L_{RTL}$ に対して，$u_0 \in [\![f]\!]_U$ ならば，$\Gamma \models f$ *(Γ は f のモデル)* である．ここで，L_{RTL} は全ての *RTL* 式の集合を意味する．

TSIS Γ が有限遷移システムならば，分枝時間時相論理によるモデル検査法とほぼ同様に，Γ が RTL 式 f を満たすかを自動的に判定できる．

命題 2 (モデル検査)
Γ の実状態が有限ならば，$\Gamma \models f$ は決定可能である．

定義 8 ($\approx_{RTL}$)

$$\Gamma_1 \approx_{RTL} \Gamma_2 \stackrel{def}{=} \forall f \in L_{RTL}.(\Gamma_1 \models f \text{ iff } \Gamma_2 \models f)$$

命題 3 (RTL と観測状態双模倣等価の関係)
任意の情報構造付き遷移システムΓ_1 と Γ_2に対して,

$$\Gamma_1 \approx_o \Gamma_2 \Leftrightarrow \Gamma_1 \approx_{RTL} \Gamma_2.$$

例 3 (RTL 式の例)
(1) $f_1 = \forall_o \forall X_o \forall_o \forall X_o \forall_o \neg(\exists X true)$
この *RTL* 式は例 *1*の囚人のジレンマを特徴づけている *(i.e.,* $\Gamma \models f_1$*)*. また, 例 *2*において, $\Gamma \approx_o \Gamma'$ であるから, $\Gamma' \models f_1$である.
(2) $f_2 = \forall_o \forall X_o(\exists_o \neg(\exists X true) \vee \exists_o \exists X true)$

3.2 RTL+

ラベル付き遷移システムの動作名および状態属性を抽象化することによって RTL を定義した. しかし, RTL はシンプルすぎて, 実際の仕様の記述には不便である. そこで, 再び情報構造を持つ遷移システムに実体としての動作名と状態属性を導入する. 以下, 動作名, 状態属性を単に実態と呼ぶ. ここでは, 実体と観測されたものを区別する. すなわち, 実体名と情報構造は独立なものとする. CCS などのモデル化においては, 実体名はラベル変更 (relabelling) によって観測名に置き換えられた. しかし, 実際の形式的定式化および検証においては, 実体名を残しかつ観測される構造を別に表現したい場合が多い. そこで, 実体および情報構造を持つ遷移システムを定義する. さらに, これをモデルとする時相論理式 RTL+を定義する. また, RTL +では, 不動点オペレータを用いた再帰的定義が記述できる.

定義 9 (実態付き遷移システム)
$T = (S, P, A, \pi, \delta)$*:* 実体付き遷移システム,
S : 実状態集合,
P : 状態属性集合,
A : 動作集合,
$\pi : S \to 2^P$ 割り当て関数,
$\delta : S \times A \to 2^S$ 非決定性遷移関数.

定義 10 (実体および情報構造付き遷移システム)
$\Gamma = (T, U, \Phi, u_0)$: 実体および情報構造付き遷移システム,
$T = (S, P, A, \pi, \delta)$: 実体付き遷移システム,
U : 観測状態集合,
$\Phi : S \to U$ 観測関数,
$u_0 \in U$: 初期観測状態.

ここでは, RTL+の構文規則を示し, 意味は簡単な例を用いて説明する.

定義 11 (RTL +)
P: 状態属性集合
A: 動作集合
Z_0*:* 観測状態変数
Z: 実状態変数
観測状態式
- Z_0 は観測状態式.
- *true* は観測状態式.
- f_1 と f_2 が観測状態式ならば, $f_1 \wedge f_2$ と $\neg f_1$ は観測状態式.
- f が 観測状態式で Z_0 が fに現れる自由観測状態変数ならば, $\mu Z_0.f$ は 観測状態式.
- h が 実状態式ならば, $\exists_o h$ は 観測状態式.

実状態式
- Z は実状態式.
- *true* は実状態式.
- $p \in P$ は 実状態式.
- h_1 と h_2 が実状態式ならば, $h_1 \wedge h_2$ と $\neg h_1$ は実状態式.
- h が実状態式で, Z が h に現れる自由実状態変数ならば, $\mu Z.h$ は 実状態式.
- g がパス式ならば, $\exists g$ は 実状態式.

パス論理式
- $a \in A$ は パス式.
- g_1 と g_2 がパス式ならば, $g_1 \wedge g_2$ と $\neg g_1$ はパス式.
- f が 観測状態式ならば, $X_o f$ は パス式.
- h が実状態式ならば, Xh は パス式.

RTL+式
- f が 観測状態式 で 自由状態変数を含まないならば, f は *RTL+* 式.

マクロ表現
- $\nu Z.f = \neg \mu Z'.\neg(\lambda Z.f)(\neg Z')$
- $[a]f = \forall(a \supset Xf)$
- $< a > f = \exists(a \wedge Xf)$

例 4 (RTL+)
$\forall_o p$: 現在のすべての可能な実状態で p は真である.
$\forall_o(\exists a)$: 現在のすべての可能な実状態で動作 a が起動できる.
$\nu Z_o.\forall_o(p \wedge \forall(a \wedge X_o Z_o))$: 動作 a で遷移する全ての観測状態の実状態で p が真である. ここで, νは最大不動点オペレータであり, 再帰的に観測状態を規定している.

$(\forall_o bird \wedge \neg\forall_o penguin) \supset \forall_o fly.$
$\forall_o\exists(on \wedge X_o\forall_o\exists(off \wedge X_o\forall_o sleep)).$
$\forall_o(p_0 \vee p_1) \wedge \exists_o p_0 \wedge \exists_o p_1.$
$\forall_o\exists(a_1 \wedge X_o\forall_o p_1)$
$\forall_o p_0 \vee \exists_o\mu Z.\exists(X_o\forall_o p_0 \vee XZ).$

4 RTL+による制御問題の定式化

リアクティブ時相論理の適用例として，リアクティブ時相論理によるリアクティブ・システムの制御問題の定式化を行なう．リアクティブ・システムの制御問題は，「制御器（プレイヤ1）」と「制御対象（プレイヤ2）」をプレイヤとする不完全情報ゲームとしてモデル化できる [9][10]．ここで，不完全情報ゲームは，以下に定義する2種類の実状態を持つ特殊な TSIS である．

定義 12 (ゲーム)
$G = (S_1, S_2, \delta_1, \delta_2)$,
S_1：プレイヤ1の実状態集合,
S_2：プレイヤ2の実状態集合,
$\delta_1 \subset S_1 \times S_2$：プレイヤ1が制御する状態遷移,
$\delta_2 \subset S_2 \times S_1$：プレイヤ2が制御する状態遷移.

図的には，プレイヤ1の実状態を6角形，プレイヤ2の実状態を4角形で表す．

定義 13 (不完全情報ゲーム)
$\Gamma = (G, U_1, U_2, \Phi_1, \Phi_2, u_0)$,
$G = (S_1, S_2, \delta_1, \delta_2)$：ゲーム,
U_1：プレイヤ1の制御時の観測状態の集合,
U_2：プレイヤ2の制御時の観測状態の集合,
$\Phi_1 : S_1 \to U_1$, S_1の観測関数,
$\Phi_2 : S_2 \to U_2$, S_2の観測関数,
$u_0 \in U_1$：初期観測状態.

例 5 (不完全情報ゲームの例)
$\Gamma = (G, \{u_{11}, u_{12}, u_{13}\}, \{u_{21}, u_{22}, u_{23}\}, \Phi_1, \Phi_2, u_{11})$,
$G = (\{s_{11}, s_{12}, s_{13}, s_{14}\}, \{s_{21}, s_{22}, s_{23}\}, \delta_1, \delta_2)$,
$\delta_1 : s_{11} \to s_{21}, s_{13} \to s_{23}, s_{12} \to s_{22}$,
$\delta_2 : s_{21} \to s_{12}, s_{21} \to s_{13}, s_{22} \to s_{11}, s_{23} \to s_{14}$,
$\Phi_1 : \Phi_1(s_{11}) = u_{11}, \Phi_1(s_{12}) = \Phi_1(s_{13}) = u_{12}, \Phi_1(s_{14}) = u_{13}$,
$\Phi_2 : \Phi_2(s_{21}) = u_{21}, \Phi_2(s_{22}) = u_{22}, \Phi_2(s_{23}) = u_{23}$.

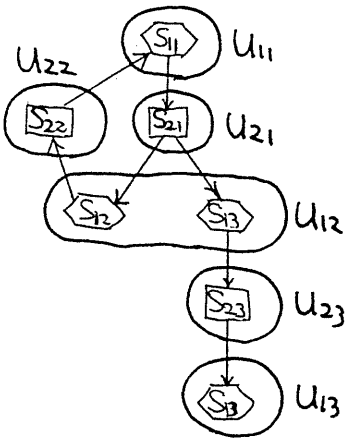


図 6: 不完全情報ゲームの例

定義 14 (ゲームの戦略)
$\pi : U_1 \to 2^{U_2}$ プレイヤ1の制御時の各観測状態に対して遷移（制御）を選ぶ関数.

リアクティブ・システムの制御とは，各時点で安全条件を満たしながら，最終的に目的条件を達成するような制御手順を実行することである．この制御手順は，ゲームにおける必勝戦略に対応する．リアクティブ・システムが制御可能（必勝戦略が存在する）か否かは，リアクティブ時相論理式で特徴づけすることが可能である．ここで，安全条件と目的条件は RTL+で表される．

命題 4 (必勝戦略の RTL+による特徴づけ)
不完全情報ゲームΓが安全条件 f_iと目的条件 f_gを達成する必勝戦略を持つための必要十分条件は，
$\Gamma \models \mu Z_o.\forall_o((f_i \wedge f_g) \vee (f_i \wedge \exists X_o\forall_o\forall X_o Z_o))$
である．

例5の不完全情報ゲームΓにおける安全条件を「デッドロック状態にならない ($f_i = \exists X true$)」，目的状態は「特になし ($f_g = false$)」とすると，$\Gamma \not\models \mu Z_o.\forall_o(\exists X true \wedge \exists X_o\forall_o\forall X_o Z_o)$ となり，Γの必勝戦略は存在しない．すなわち，制御不可能である．これは，s_{12}と s_{13}が制御器にとって識別できないことに原因がある．これを，識別できるようにゲームを改良する（図7）との必勝戦略 ($\pi(u_{11}) = \{u_{21}\}, \pi(u_{12}) = \{u_{22}\}$) が存在し，制御可能となる．

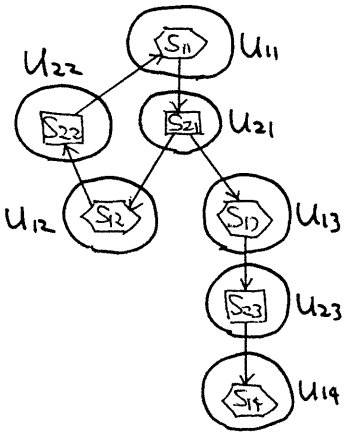


図 7: 改良したゲーム

5 おわりに

リアクティブ・システムの観測可能性を明示的に記述できる形式的記述言語であるリアクティブ時相論理を提案した．今回の提案は，リアクティブ性のある一部を定式化したものである．今後の当面の課題は，リアクティブ・システムのもう1つの本質である制御可能性をより明示的に扱えるように論理を洗練化することである．また，観測不可能な内部遷移の扱いについても洗練化していく．さらに，今回提案した情報構造はリアクティブ・システムの制御器という1つの視点からの観測可能性に注目していたが，これを複数の視点からの観測可能性に拡張し，マルチエージェントシステムの形式的記述言語に発展させ，「協調アーキテクチャ」の時相論理による定式化に適用していきたい．

謝辞: 本研究は，次世代産業基盤技術開発「新ソフトウェア構造化モデルの研究開発」の一環として情報処理振興事業協会が新エネルギー・産業技術総合開発機構からの委託をうけて実施したものである．

参考文献

[1] 松本 他：時相論理とその応用，情報処理 Vol.30, No.6 (1989).

[2] P. Wolper : On the relation of programs and computations to models of temporal logic, LNCS 398 (1989).

[3] Z. Manna and A. Pnueli : The Temporal Logic of Reactive and Concurrent Systems - Specification -, Springer-Verlag (1992).

[4] J. S. Ostroff : Temporal Logic for Real-time Systems, John Willey & Sons Inc. (1989).

[5] R. Alur, T.A. Henzinger : A Really Temporal Logic, 30th FOCS (1989).

[6] R. Milner : Communication and Concurrency, Prentice-Hall International (1989).

[7] E.M. Clarke, E.A. Emerson, A.P. Sistla : Automatic Verification of Finite-state Concurrent Systems Using Temporal Logic Specifications, ACM TOPLAS, 8, 2 (1986).

[8] 鈴木光男：ゲーム理論入門，共立出版 (1981).

[9] 内平, 本位田：ゲーム理論による協調エージェントのモデル化，情報処理学会研究報告 PRG3-7 (1991).

[10] 内平, 本位田：ゲーム理論を用いたリアクティブ・ペトリネットの制御手順生成，電気学会 システム・制御研究会 SC-91-13 (1991).