時間模倣関係による実時間システムの階層的設計手法

山根 智†

通信プロトコルや制御システムなどの実時間システムはリアクティブ性が存在して，大規模化している．さらに，システムの論理動作の正当性だけでなく，実時間性の正当性が非常に重要である．ゆえに，実時間システムでは，階層的に設計できて，それらの正当性が保証できることが重要である．本論文では，要求仕様から設計仕様までを，同一の形式的仕様記述言語で記述して，それらの間の整合性を自動検証できる手法を提案する．とりわけ，実時間性と公平性を考慮した実時間システムの階層的な設計に着目する．具体的には，非決定性時間オートマトンによる統一的な仕様記述および公平性を考慮した時間模倣関係，すなわち∃-時間模倣関係，∀-時間模倣関係による整合性の保証である．最後に，簡単な事例により，提案した階層的な設計手法の有効性を示す．

A Hierarchical Design Method of Real-time Systems Based on Timed Simulation Relation

SATOSHI YAMANE†

As real-time systems such as communication protocols and control systems are reactive and larger, it is difficult to design real-time systems. For this reason, a hierarchical design method is useful. In the hierarchical design method, it is important to verify whether lower levels satisfy upper levels or not. In general, the language inclusion verification method is useful for verifying it. But, as nondeterministic timed automata are not closed under complementation, it is impossible to use the language inclusion verification method.

In this paper, we propose the hierarchical design method based on timed simulation method. Especially, we generalize existing timed simulation methods and propose *safety* timed simulation relation and ∃-timed simulation relation, ∀-timed simulation relation. Finally, we show the proposed method is effective by some example.

1. まえがき

通信プロトコルや制御システムなどの実時間システムはリアクティブ性が存在して，大規模化している．さらに，システムの論理動作の正当性だけでなく，実時間性の正当性が非常に重要である．ゆえに，実時間システムでは，階層的に設計できて，それらの正当性が保証できることが重要である．本論文では，要求仕様から設計仕様までを，同一の形式的仕様記述言語で記述して，それらの間の整合性を自動検証できる手法を提案する．とりわけ，実時間性と公平性の仕様記述および階層間の整合性検証に着目する．公平性とは，プロセスが無限に動作することを意味する数学的抽象化であり，プロセスの物理的な実行速度が抽象化できたり，プロセスの公平な実行を仮定する有益な概念である[1]．無限語上の ω-オートマトン[2]で仕様記述すれば公平性を有するプロセスが表現でき，通常の有限語上の有限オートマトンで仕様記述すれば公平性を有するプロセスが表現できない．特に，メッセージ損失のある通信プロトコルなどのように，公平性を仮定しないと仕様記述することが困難なシステムが存在する．なぜならば，ω-オートマトンで仕様記述するときに，メッセージを入力する状態と出力する状態を受理状態集合として仕様記述するならば，無限の入力データを受け付けて，有限のメッセージ損失が存在しても，いつかは無限のデータを出力することが表現できるからである．このために，公平性は実時間システムにとって非常に重要である．

従来の実時間システムの形式的検証の研究としては，時間オートマトンの言語包含検証[3]やモデルチェッキング検証[4]，プロセス代数の等価性判定[5]，離散時間で拡張した時間LOTOSの双模倣等価性判定[6]などがある．これらは，階層間の整合性関係が不十分であったり，実時間性が表現できない．

階層間の整合性を検証できる関係としては，言語包

† 鹿児島大学工学部情報工学科
Department of Information and Computer Science, Kagoshima University

含関係や双模倣関係，模倣関係などが知られている．

(1) 言語包含検証は正則性や公平性を含む安全性や活性が検証できる強力かつシンプルな検証手法である．しかし，時間オートマトンの言語包含検証においては，検証仕様を非決定性時間オートマトンで記述すれば，言語包含検証は不可能である．なぜならば，非決定性時間オートマトンの補集合演算の閉包性がないために，時間オートマトンの言語包含検証は決定不能問題になるからである[3]．

(2) 双模倣関係では，上位レベルの仕様と下位レベルの仕様との動作の違いを区別できないとき，それらを等価なものとして扱う．しかし，仕様を階層的かつ増補的に開発していく場合には，下位レベルの仕様には，上位レベルの仕様には含まれていない例外処理などが含まれる．このために，双模倣関係は適切ではない．

以上より，階層間の整合性を検証できる関係として，我々は模倣関係を使用する．模倣関係の検証を目的とした従来の代表的な研究には，以下のものがある．

(1) MIT の Lynch らは，分散アルゴリズムを対象として，模倣関係が検証できる公理系を開発した[7]．さらに，Lynch らは，実時間システムを対象として，時間模倣関係が検証できる公理系を開発した[8]．しかし，これらは，自動化されていない．

(2) スタンフォード大学の Dill らは，公平性を追加した模倣関係に対して，二分決定グラフによる自動的なシンボリック検証手法を提案した[9]．しかし，実時間システムの検証ができない．

そこで，本論文では，実時間システムを対象として，公平性と実時間性の存在する模倣関係を一般化して，それらの概念を提案する．

本論文では，要求仕様から設計仕様までを，非決定性時間オートマトンで統一的に仕様記述して，公平性と実時間性の存在する模倣関係により，仕様の間の整合性を自動検証できる階層的な設計手法を提案する．

以下の本論文は，2章では実時間システムの仕様記述手法を導入し，3章では時間模倣関係を基礎とした階層的な設計手法を提案する．4章では設計支援システムと設計事例を示し，5章ではまとめを述べる．

2. 実時間システムの仕様記述手法と時間模倣関係

2.1 実時間システムの仕様記述手法

実時間システムの要求仕様や設計仕様は非決定性時間オートマトン[3]で記述する．時間オートマトンは ω-オートマトンをクロック変数によるタイミング制約式により拡張したものであり，停止性が仮定できないプロセスの記述を可能とする．

Definition 1（非決定性時間オートマトンの定義）非決定性時間オートマトンは $A=(\Sigma, S, S_0, C, E, F)$ の6つ組で定義する．

ここで，

Σ：有限イベント集合

S：有限状態集合

$S_0 \subseteq S$：初期状態集合

C：クロックの有限集合

$E \subseteq S \times S \times \Sigma \times 2^C \times \Phi(C)$：遷移関係

$F \subseteq S$ は Buchi 型の受理状態集合

$\Phi(C)$ はクロック集合 C のタイミング制約式 δ であり，クロック集合 C ($x \in C$) と非負整数の時刻定数 d により以下のように帰納的に定義される：

(1) $x \leq d$ や $d \leq x$ は δ である．

(2) δ_1 と δ_2 が δ ならば，$\neg\delta_1$ や $\delta_1 \wedge \delta_2$ は δ である．

状態遷移 $(s, s\prime, a, \lambda, \delta)$ はイベント a による状態 s から状態 $s\prime$ までの遷移を表す．集合 $\lambda \subseteq C$ は，この状態遷移でリセットされるクロック集合を表す．以下，本論文では，

$$s \xrightarrow{a,\lambda,\delta} s\prime$$

は，状態遷移 $(s, s\prime, a, \lambda, \delta) \in E$ を意味する．ただし，λ が存在しない場合は，

$$s \xrightarrow{a,\delta} s\prime$$

などのように記述する．非決定性時間オートマトンは時間付き言語 (σ, τ) を受理する．ここで，$\sigma = \sigma_1, \sigma_2, \ldots$，$\tau = \tau_1, \tau_2, \ldots$ である．ただし，$\sigma_i \in \Sigma$, $\tau_i \in \mathbf{R}$（$\mathbf{R}$：非負の実数）とする．非決定性時間オートマトンの走査列 r は，以下のような無限列である：

$$\langle s_0, \nu_0 \rangle \xrightarrow{\sigma_1,\tau_1} \langle s_1, \nu_1 \rangle \xrightarrow{\sigma_2,\tau_2} \langle s_2, \nu_2 \rangle \xrightarrow{\sigma_3,\tau_3} \ldots$$

ここで，$s_i \in S$, $\nu_i \in [C \rightarrow \mathbf{R}]$ である．走査列 r の中で無限回出現する状態集合を $inf(r)$ とする．$inf(r) \cap F \neq \phi$ のときに限り，走査列 r は受理走査列である． □

本論文で対象とする非決定性時間オートマトンは，以下のように強力な仕様記述言語である．

(1) 本論文の非決定性時間オートマトンは，正確には非決定性時間 Buchi オートマトンであり，すべての時間付き正則 ω 言語が表現できる．一

方，決定性時間 Buchi オートマトンには受理できない，ある種の時間付き正則 ω 言語が存在する．

(2) 非決定性時間オートマトンは，決定性時間オートマトンの抽象化であり，同一の仕様を記述する場合には，非決定性時間オートマトンのほうが少ない状態数で仕様記述できる．

以上より，非決定性時間オートマトンによって，すべての階層の仕様を統一的に記述することはおおいに意味がある．

非決定性時間オートマトンの意味は，時間付き受理言語で定義できる．その詳細は文献 3) に定義されているので，紙面の都合上から省略する．

次に，非決定性時間オートマトンの例を示す．

Example 1（**非決定性時間オートマトンの例**）非決定性時間オートマトンの例を図 **1** に示す．図中では，有向辺のラベルにはイベント a とリセット式 λ，タイミング制約式 δ がある．$s_0 \to s_1$ の遷移はイベント a により生起して，クロック変数 y がリセットされる．また，$s_1 \to s_2$ の遷移はイベント b が生起し，かつクロック変数が条件 $y = 1$ を満たすとき起きる．さらに，$s_1 \to s_3$ の遷移はイベント b が生起し，かつクロック変数が条件 $y < 2$ を満たすとき起きる．ゆえに，状態 s_1 において，イベント b が生起し，かつクロック変数が条件 $y = 1$ を満たすならば，状態 s_2 および s_3 に遷移する可能性がある．すなわち，このオートマトンは非決定性である．なお，受理状態集合は，$F = \{s_3\}$ である． □

実時間システムは多くの非決定性時間オートマトンから構成されていると考えて，実時間システムの仕様は非決定性時間オートマトンのカルテジアン積で表現する．非決定性時間オートマトンの積や和などの演算閉包性は文献 3) で定義しているので，本論文では紙面の都合上から省略する．以下では，非決定性時間オートマトンのカルテジアン積を定義する．

Definition 2（**非決定性時間オートマトンのカルテジアン積**）非決定性時間オートマトン $A\prime = (\Sigma, S\prime, S_0\prime, C\prime, E\prime, F\prime)$ と $A\prime\prime = (\Sigma, S\prime\prime, S_0\prime\prime, C\prime\prime, E\prime\prime, F\prime\prime)$ のカルテジアン積 $A\prime \parallel A\prime\prime = (\Sigma, S, S_0, C, E, F)$ を定義する．

(1) $S = S\prime \times S\prime\prime$

(2) $S_0 = S_0\prime \times S_0\prime\prime$

(3) $C = C\prime \cup C\prime\prime$

(4) $(s\prime, t\prime, a, \lambda\prime, \delta\prime) \in E\prime$ と $(s\prime\prime, t\prime\prime, b, \lambda\prime\prime, \delta\prime\prime) \in E\prime\prime$ に対して，E を以下のように定義する．ただし，$(s, t, a, \lambda, \delta)$ はイベント a による状態 s から状態 t までの遷移を表して，集合 $\lambda \subseteq C$ は，この状態遷移でリセットされるクロック集合を表し，δ はタイミング制約式を表す．

(a) $a = b$ のとき

$(s\prime \times s\prime\prime, t\prime \times t\prime\prime, a, \lambda\prime \cup \lambda\prime\prime, \delta\prime \wedge \delta\prime\prime) \in E$

(b) $a \neq b$ のとき

$(s\prime \times s\prime\prime, t\prime \times s\prime\prime, a, \lambda\prime, \delta\prime) \in E$

または

$(s\prime \times s\prime\prime, s\prime \times t\prime\prime, b, \lambda\prime\prime, \delta\prime\prime) \in E$

(5) $F = F\prime \times F\prime\prime$ □

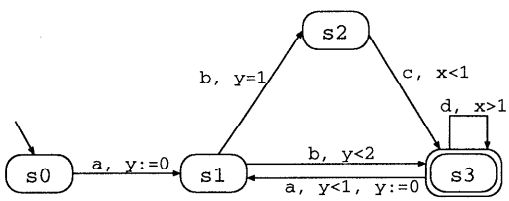

図 1 非決定性時間オートマトンの例

Fig. 1 Example of nondeterministic timed automaton.

2.2 時間模倣関係の定義

次に，非決定性時間オートマトンの上で，以下の 3 種類の時間模倣関係を提案する．

(1) 公平性を考慮しない安全性時間模倣関係

(2) ある状態列で公平性を満たす ∃-活性時間模倣関係

(3) すべての状態列で公平性を満たす ∀-活性時間模倣関係

Definition 3（**安全性時間模倣関係の定義**）上位レベルの仕様 $A\prime = (\Sigma, S\prime, S_0\prime, C\prime, E\prime, F\prime)$ と下位レベルの仕様 $A\prime\prime = (\Sigma, S\prime\prime, S_0\prime\prime, C\prime\prime, E\prime\prime, F\prime\prime)$ を任意の非決定性時間オートマトンとする．次の条件を満たす $R \subseteq S\prime \times S\prime\prime$ を $A\prime$ から $A\prime\prime$ への安全性時間模倣関係と呼び，そのような関係 R が存在するとき，$A\prime \preceq A\prime\prime$ と記述する．なお，安全性時間模倣関係では，$A\prime$ と $A\prime\prime$ において，$S\prime = F\prime$ および $S\prime\prime = F\prime\prime$ と考えて，受理条件に無関係に無限に動作すると考える．

(1) 安全性時間模倣関係

$(s_i\prime, s_i\prime\prime) \in R$ ならば $\forall \sigma \in \Sigma, \lambda, \delta$ について，

$$s_i\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime$$

である $s_{i+1}\prime$ が存在するならば，

$$s_i\prime\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime\prime$$

である $s_{i+1}\prime\prime$ が存在して，$(s_{i+1}\prime, s_{i+1}\prime\prime) \in R$ である．なお，$s_i\prime, s_{i+1}\prime \in S\prime$ かつ $s_i\prime\prime, s_{i+1}\prime\prime \in S\prime\prime$ である．

(2) 初期条件

$\forall s_0\prime \in S_0\prime$ に対して，$(s_0\prime, s_0\prime\prime) \in R$ を満たす

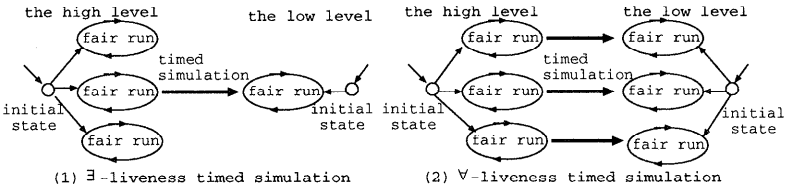

図2 活性時間模倣関係のイメージ
Fig. 2 Image of live timed simulation relation.

$s_0\prime\prime \in S_0\prime\prime$ が存在する．ここで，$(s_0\prime, s_0\prime\prime) \in R$ とは，上位レベルの初期状態から下位レベルの初期状態への安全性時間模倣関係が存在することを意味する．すなわち，安全性時間模倣関係の中に，初期状態が含まれていることを意味する． □

Definition 4（∃-活性時間模倣関係の定義）上位レベルの仕様 $A\prime = (\Sigma, S\prime, S_0\prime, C\prime, E\prime, F\prime)$ と下位レベルの仕様 $A\prime\prime = (\Sigma, S\prime\prime, S_0\prime\prime, C\prime\prime, E\prime\prime, F\prime\prime)$ を任意の非決定性時間オートマトンとする．次の条件を満たす $R \subseteq S\prime \times S\prime\prime$ を $A\prime$ から $A\prime\prime$ への ∃-活性時間模倣関係と呼び，$A\prime \preceq^{\exists} A\prime\prime$ と記述する．

(1) ∃-活性時間模倣関係

$A\prime$ の $inf(r\prime) \cap F\prime \neq \phi$ を満たすある状態列 $r\prime = s_0\prime, s_1\prime, \ldots$ に対して，$A\prime\prime$ の $inf(r\prime\prime) \cap F\prime\prime \neq \phi$ を満たす状態列 $r\prime\prime = s_0\prime\prime, s_1\prime\prime, \ldots$ が存在して，以下の条件を満たす．

$(s_i\prime, s_i\prime\prime) \in R$ ならば $\forall\sigma \in \Sigma, \lambda, \delta$ について，

$$s_i\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime$$

である $s_{i+1}\prime$ が存在するならば，

$$s_i\prime\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime\prime$$

である $s_{i+1}\prime\prime$ が存在して，$(s_{i+1}\prime, s_{i+1}\prime\prime) \in R$ である．なお，$s_i\prime, s_{i+1}\prime \in S\prime$ かつ $s_i\prime\prime, s_{i+1}\prime\prime \in S\prime\prime$ である．

(2) 初期条件

$\forall s_0\prime \in S_0\prime$ に対して，$(s_0\prime, s_0\prime\prime) \in R$ を満たす $s_0\prime\prime \in S_0\prime\prime$ が存在する．ここで，$(s_0\prime, s_0\prime\prime) \in R$ とは，上位レベルの初期状態から下位レベルの初期状態への ∃-活性時間模倣関係が存在することを意味する．すなわち，∃-活性時間模倣関係の中に，初期状態が含まれていることを意味する． □

Definition 5（∀-活性時間模倣関係の定義）上位レベルの仕様 $A\prime = (\Sigma, S\prime, S_0\prime, C\prime, E\prime, F\prime)$ と下位レベルの仕様 $A\prime\prime = (\Sigma, S\prime\prime, S_0\prime\prime, C\prime\prime, E\prime\prime, F\prime\prime)$ を任意の非決定性時間オートマトンとする．次の条件を満たす $R \subset S\prime \times S\prime\prime$ を $A\prime$ から $A\prime\prime$ への ∀-活性時間模倣関係と呼び，$A\prime \preceq^{\forall} A\prime\prime$ と記述する．

(1) ∀-活性時間模倣関係

$A\prime$ の $inf(r\prime) \cap F\prime \neq \phi$ を満たすすべての状態列 $r\prime = s_0\prime, s_1\prime, \ldots$ ごとに，$A\prime\prime$ の $inf(r\prime\prime) \cap F\prime\prime \neq \phi$ を満たす状態列 $r\prime\prime = s_0\prime\prime, s_1\prime\prime, \ldots$ が存在して，以下の条件を満たす．

$(s_i\prime, s_i\prime\prime) \in R$ ならば $\forall\sigma \in \Sigma, \lambda, \delta$ について，

$$s_i\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime$$

である $s_{i+1}\prime$ が存在するならば，

$$s_i\prime\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime\prime$$

である $s_{i+1}\prime\prime$ が存在して，$(s_{i+1}\prime, s_{i+1}\prime\prime) \in R$ である．なお，$s_i\prime, s_{i+1}\prime \in S\prime$ かつ $s_i\prime\prime, s_{i+1}\prime\prime \in S\prime\prime$ である．

(2) 初期条件

$\forall s_0\prime \in S_0\prime$ に対して，$(s_0\prime, s_0\prime\prime) \in R$ を満たす $s_0\prime\prime \in S_0\prime\prime$ が存在する．ここで，$(s_0\prime, s_0\prime\prime) \in R$ とは，上位レベルの初期状態から下位レベルの初期状態への ∀-活性時間模倣関係が存在することを意味する．すなわち，∀-活性時間模倣関係の中に，初期状態が含まれていることを意味する． □

安全性時間模倣関係および ∃-活性時間模倣関係，∀-活性時間模倣関係は以下のように異なる．安全性時間模倣関係は受理条件に無関係な時間模倣関係なので，活性時間模倣関係よりも弱い関係である．次に，∃-活性時間模倣関係と ∀-活性時間模倣関係の定義のイメージを**図 2** に示す．∃-活性時間模倣関係では，上位レベルの仕様の公平な状態列に対して，下位レベルの仕様の公平な状態列が 1 つでも存在すればよい．一方，∀-活性時間模倣関係では，上位レベルの仕様のすべての公平な状態列に対して，下位レベルの仕様の公平な状態列が存在しなければならない．

3. 時間模倣関係による階層的な設計手法

本論文では，言語包含関係よりも強い関係である模

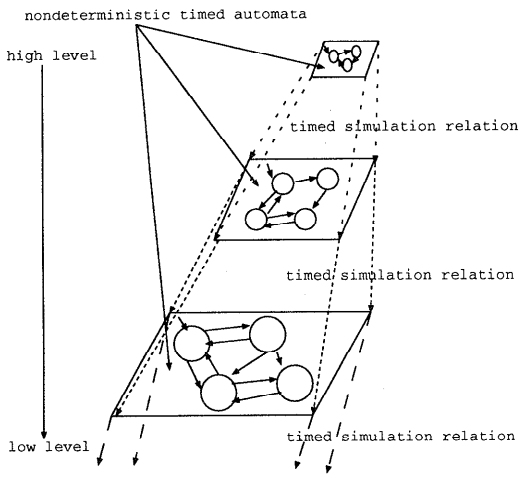

図3 階層的設計手法のイメージ図
Fig. 3 Image of hierarchical design method.

倣関係を基礎として，公平性を考慮した時間模倣関係および非決定性時間オートマトンによる統一的な仕様記述を基礎とした実時間システムの階層的な設計手法を提案する．なお，非決定性時間オートマトンの補集合演算を回避できる時間模倣関係により，統一的な仕様記述言語による階層的な設計が実現できる．

3.1 階層的な設計手法

我々は，実時間システムの抽象度の高い上位レベルの仕様と抽象度の低い下位レベルの仕様を同一の非決定性時間オートマトンで仕様記述して，時間模倣関係によりそれらの間の整合性を自動検証できる階層的な設計手法を提案する．

Definition 6（**階層的設計手法**）実時間システムの階層的設計手法は次の手順から構成される．そのイメージを図 **3** に示す．

(1) まず，実時間システムの上位レベルの仕様を非決定性時間オートマトンで記述する．
(2) 次に，上位レベルの仕様を基礎として，それを実現する下位レベルの仕様を非決定性時間オートマトンで設計する．
(3) 最後に，上位レベルの仕様から下位レベルの仕様への時間模倣関係が存在するまで，下位レベルの仕様を修正する．
(4) 以上のステップを繰り返して，最終的な実現仕様を設計する． □

3.2 時間模倣関係による階層間の整合性の検証手法

時間模倣関係による整合性の検証では，非決定性時間オートマトン上の時間模倣関係の存在性問題はリージョングラフ上のリージョン模倣関係の存在性問題に帰着できる．まず，リージョングラフを定義して，リージョン模倣関係を定義する．次に，リージョン模倣関係の存在性問題への帰着性を示して，その検証アルゴリズムを示す．

3.2.1 リージョングラフの構成

非決定性時間オートマトンのタイミング制約記述の時間領域は稠密なので，クロック変数にそのまま時間を割り当てると，無限個の状態とクロック値のペアができてしまう．しかし，タイミング制約式の中の定数部分（たとえば $x < d$ の d の部分）が非負整数なので，クロック値の整数部分と小数部分の順序関係が同じならば，非決定性時間オートマトンの動作は区別されない．この考えから，我々は，動作が区別されないクロック割当ての同値関係により，クロックリージョンと呼ばれる同値類を構成する．そして，クロックリージョンにより，非決定性時間オートマトンから，有限な商構造であるリージョングラフを構成することができる[3]．

まず，このクロック割当ての同値関係を定義する．

Definition 7（**クロック割当ての同値関係**）非決定性時間オートマトン A が与えられたとする．任意のクロック変数 $x \in C$ に対して，A に現れる最大のクロック定数を c_x とする．また，時間 $t \in R$ に対して，

fract(t)：t の小数部分
$\lfloor t \rfloor$：t の整数部分

とする．A のクロック割当ての集合を $\Gamma(A)$ とし，その要素を $\nu, \nu\prime \in [C \to R]$ とする．次の条件を満たすときに限り，$\nu \equiv \nu\prime$ と表す．

(1) 任意のクロック変数 $x \in C$ に対して，$\lfloor \nu(x) \rfloor$ と $\lfloor \nu\prime(x) \rfloor$ が同じ，または，$\nu(x)$ と $\nu\prime(x)$ の両方が c_x よりも大きい．
(2) $\nu(x) \le c_x$ と $\nu(y) \le c_y$ であるすべての $x, y \in C$ に対して，$fract(\nu\prime(x)) \le fract(\nu\prime(y))$ のときに限り，$fract(\nu(x)) \le fract(\nu(y))$ である．
(3) $\nu(x) \le c_x$ であるすべての $x \in C$ に対して，$fract(\nu\prime(x)) = 0$ のときに限り，$fract(\nu(x)) = 0$ である． □

非決定性時間オートマトン A のクロックリージョンは同値関係 $\equiv$ によって導かれるクロック割当ての同値類であり，$[\nu]$ と記述する．

次に，クロックリージョンを基礎として，非決定性時間オートマトンのリージョングラフを定義する．

Definition 8（**リージョングラフ（商構造）**）非決定性時間オートマトン $A = (\Sigma, S, S_0, C, E, F)$ のリージョングラフは $G = (Q, Q_0, \Sigma, E\prime, F\prime)$ で定義される．ここで，

(1) Q：$\langle s,[\nu]\rangle$ の有限集合
  (a) $s \in S$
  (b) $[\nu]=\{\nu\prime|\nu \equiv \nu\prime\}$, $\nu \in \Gamma(A)$
(2) Q_0：$\langle s_0,[\nu_0]\rangle$ の有限集合
  (a) $s_0 \in S_0$
  (b) $[\nu_0]$：すべてのクロック変数が 0 の順序対の集合
(3) Σ：有限イベント集合
(4) $E\prime \subseteq Q \times \Sigma \times Q$：状態遷移関係
(5) $F\prime$：$\forall s \in F$に対する $\langle s,[\nu]\rangle$ の有限集合 □

Example 2（リージョングラフ構成の事例）図 4 では，(1) 非決定性時間オートマトンから (2) クロックリージョンを構成し，クロックリージョンを基礎として (3) リージョングラフを構成する．

(1) 非決定性時間オートマトンについて
初期状態 s_0 において，$a, y=z=1$ のとき，2 つの遷移 $s_0 \to s_2$，$s_0 \to s_1$ が可能なので，これは非決定性時間オートマトンである．非決定性時間オートマトンにより，少ない状態数で，仕様記述が可能となった．

(2) クロックリージョンについて
クロック変数 y, z の時間定数が 0 および 1 なので，クロックリージョンは以下のように 16 通りである：$[y=0,z=0]$, $[y=0,0<z<1]$, $[y=0,z=1]$, $[y=0,z>1]$, $[0<y<1,z=0]$, $[0<y<1,0<z<1]$, $[0<y<1,z=1]$, $[0<y<1,z>1]$, $[y=1,z=0]$, $[y=1,0<z<1]$, $[y=1,z=1]$, $[y=1,z>1]$, $[y>1,z=0]$, $[y>1,0<z<1]$, $[y>1,z=1]$, $[y>1,z>1]$.
なお，クロックリージョン $[\nu]$ は，クロック変数 y，z がとりうる時間領域の順序対で表現される．

(3) リージョングラフについて
  (a) s_0 について
  状態遷移 $s_0 \to s_1$ と $s_0 \to s_2$ が存在するので，状態 s_0 のリージョングラフは 2 つに分離する．
    (i) $s_0 \to s_1$ の時間制約が $y=z\leq 1$ なので，s_0 のクロックリージョンは，$[y=0,z=0]$ と $[0<y<1,0<z<1]$, $[y=1,z=1]$ の和集合 $[0\leq y=z\leq 1]$ である．
    (ii) $s_0 \to s_2$ の時間制約が $y=z\geq 1$ なので，s_0 のクロックリージョンは，$[y=0,z=0]$ と $[0<y<1,0<z<1]$, $[y=1,z=1]$, $[y>1,z>1]$ の和集合 $\{[0\leq y=z\leq 1]\}\cup\{[y=z>1]\}$ である．
  (b) s_1 について
  $s_0 \to s_1$ の時間制約が $y=z\leq 1$ であり，リセット式 $y:=0$，および $s_1 \to s_2$ の時間制約が $y=1\leq z$ であるので，s_1 のクロックリージョンは，$[y=0,0<z<1]$ と $[y=0,z=1]$, $[y=0,z>1]$, $[0<y<1,0<z<1]$, $[0<y<1,z=1]$, $[0<y<1,z>1]$, $[y=1,z>1]$ の和集合 $\{[0\leq y\leq 1]\}\cap\{[y\geq z]\}$ である．
  (c) s_2 について
  $s_1 \to s_2$ の時間制約が $y=1\leq z$ および $s_0 \to s_2$ の時間制約が $y=z\geq 1$ であるので，s_2 のクロックリージョンは，$[z\geq y>1]$ である． □

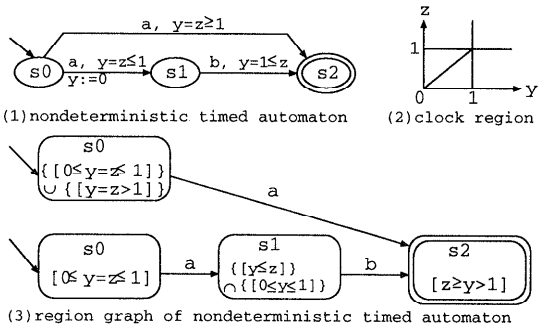

図 4 リージョングラフ構成の例
Fig. 4 Example of region graph construction.

3.2.2 リージョン模倣関係の存在性問題への帰着性

非決定性時間オートマトン上の時間模倣関係の存在性問題はリージョングラフ上のリージョン模倣関係の存在性問題に帰着できることを示す．

まず，上位レベルの仕様 $A\prime=(\Sigma,S\prime,S_0\prime,C\prime,E\prime,F\prime)$ と下位レベルの仕様 $A\prime\prime=(\Sigma,S\prime\prime,S_0\prime\prime,C\prime\prime,E\prime\prime,F\prime\prime)$ に対して，以下を定義する．

(1) $RG_{A\prime\|A\prime\prime}=(Q,Q_0,\Sigma,E,F)$ は，$A\prime \parallel A\prime\prime$ のリージョングラフとする．ここで，
  (a) Q：$\langle s\prime\times s\prime\prime,[\nu]\rangle$ の有限集合
    (i) $s\prime \in S\prime$, $s\prime\prime \in S\prime\prime$
    (ii) $[\nu]=\{\nu\prime|\nu\equiv\nu\prime\}$, $\nu\in\Gamma(A\prime\parallel A\prime\prime)$
  (b) Q_0：$\langle s_0\prime\times s_0\prime\prime,[\nu_0]\rangle$ の有限集合
    (i) $s_0\prime \in S_0\prime$, $s_0\prime\prime \in S_0\prime\prime$
    (ii) $[\nu_0]$：すべてのクロック変数が 0

の順序対の集合

(c) Σ：有限イベント集合

(d) $E \subseteq Q \times \Sigma \times Q$：状態遷移関係

(e) F：$\forall s\prime \in F\prime$ と $\forall s\prime\prime \in F\prime\prime$ に対する $\langle s\prime \times s\prime\prime, [\nu]\rangle$ の有限集合

(2) $RG(s\prime, s\prime\prime)$ は，$\langle s\prime \times s\prime\prime, [\nu]\rangle$ または $\{\langle s\prime, [\nu]\rangle, \langle s\prime\prime, [\nu]\rangle\}$ であり，状態の対 $(s\prime, s\prime\prime)$ が属するリージョングラフの同値類を表す．

以下に，リージョングラフ上の3種類のリージョン模倣関係を定義する．

Definition 9（安全性リージョン模倣関係の定義） 任意の $RG(s_i\prime, s_i\prime\prime) \in \chi$ に対して，以下の条件が満たされるときに限り，$\chi \subseteq RG_{A\prime \| A\prime\prime}$ は，$A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係であるという：

(1) 任意の $\sigma \in \Sigma$ に対して，

$$\langle s_i\prime, [\nu_i]\rangle \xrightarrow{\sigma} \langle s_{i+1}\prime, [\nu_{i+1}]\rangle$$

ならば，$RG(s_{i+1}\prime, s_{i+1}\prime\prime) \in \chi$ である $\langle s_{i+1}\prime\prime, [\nu_{i+1}]\rangle$ に対して，

$$\langle s_i\prime\prime, [\nu_i]\rangle \xrightarrow{\sigma} \langle s_{i+1}\prime\prime, [\nu_{i+1}]\rangle$$

である．

(2) $\forall s_0\prime \in S_0\prime$ に対して，$RG(s_0\prime, s_0\prime\prime) \in \chi$ を満たす $s_0\prime\prime \in S_0\prime\prime$ が存在する． □

Theorem 1（安全性時間模倣関係と安全性リージョン模倣関係） $RG(s_i\prime, s_i\prime\prime) \in \chi$ に対して，$R_\chi = \{(s_i\prime, s_i\prime\prime) | RG(s_i\prime, s_i\prime\prime) \in \chi\}$ とする．χ が $A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係である必要十分条件は，R_χ が $A\prime$ から $A\prime\prime$ への安全性時間模倣関係 $A\prime \preceq A\prime\prime$ である．

（証明の方針）

以下の2つの場合に分けて証明する．

(1) R_χ が $A\prime$ から $A\prime\prime$ への安全性時間模倣関係ならば，χ は $A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係であることの証明：

R_χ が $A\prime$ から $A\prime\prime$ への安全性時間模倣関係であると仮定する．定義より明らかに，R_χ が $A\prime$ から $A\prime\prime$ への安全性時間模倣関係ならば，χ は $A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係である．

(2) χ が $A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係ならば，R_χ は $A\prime$ から $A\prime\prime$ への安全性時間模倣関係であることの証明：

χ が $A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係であると仮定する．ある σ，λ，δ に対して，$(s_i\prime, s_i\prime\prime) \in R_\chi$ かつ

$$s_i\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime$$

とする．

$$s_i\prime\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime\prime$$

かつ $(s_{i+1}\prime, s_{i+1}\prime\prime) \in R_\chi$ であるような $s_{i+1}\prime\prime$ が存在することを示せばよい．ここで，本来の非決定性時間オートマトン $A\prime$, $A\prime\prime$ の σ に対する，$s_i\prime$ から $s_{i+1}\prime$ への状態遷移および $s_i\prime\prime$ から $s_{i+1}\prime\prime$ への状態遷移は，以下のように定義できる：

$$s_i\prime \xrightarrow{\sigma,\lambda_i\prime,\delta_i\prime} s_{i+1}\prime$$

および

$$s_i\prime\prime \xrightarrow{\sigma,\lambda_i\prime\prime,\delta_i\prime\prime} s_{i+1}\prime\prime.$$

そして，ある σ，λ，δ を $\sigma, \lambda_i\prime \cup \lambda_i\prime\prime, \delta_i\prime \wedge \delta_i\prime\prime$ とする．このときには，明らかに，

$$s_i\prime\prime \xrightarrow{\sigma,\lambda,\delta} s_{i+1}\prime\prime$$

かつ $(s_{i+1}\prime, s_{i+1}\prime\prime) \in R_\chi$ であるような $s_{i+1}\prime\prime$ が存在する．

以上より，χ が $A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係ならば，R_χ は $A\prime$ から $A\prime\prime$ への安全性時間模倣関係である． □

次に，安全性リージョン模倣関係と同様に，∃-活性リージョン模倣関係および ∀-活性リージョン模倣関係を定義する．

Definition 10（∃-活性リージョン模倣関係の定義） $A\prime$ の $inf(r\prime) \cap F\prime \neq \phi$ を満たすある状態列 $r\prime = s_0\prime, s_1\prime, \ldots$ に対して，$A\prime\prime$ の $inf(r\prime\prime) \cap F\prime\prime \neq \phi$ を満たす状態列 $r\prime\prime = s_0\prime\prime, s_1\prime\prime, \ldots$ が存在する場合に，$r\prime$ と $r\prime\prime$ に関して，$RG(s_i\prime, s_i\prime\prime) \in \chi$ に対し，以下の条件が満たされるときに限り，$\chi \subseteq RG_{A\prime \| A\prime\prime}$ は，$A\prime$ から $A\prime\prime$ への ∃-活性リージョン模倣関係であるという：

(1) 任意の $\sigma \in \Sigma$ に対して，

$$\langle s_i\prime, [\nu_i]\rangle \xrightarrow{\sigma} \langle s_{i+1}\prime, [\nu_{i+1}]\rangle$$

ならば，$RG(s_{i+1}\prime, s_{i+1}\prime\prime) \in \chi$ である $\langle s_{i+1}\prime\prime, [\nu_{i+1}]\rangle$ に対して，

$$\langle s_i\prime\prime, [\nu_i]\rangle \xrightarrow{\sigma} \langle s_{i+1}\prime\prime, [\nu_{i+1}]\rangle$$

である．

(2) $\forall s_0\prime \in S_0\prime$ に対して，$RG(s_0\prime, s_0\prime\prime) \in \chi$ を満たす $s_0\prime\prime \in S_0\prime\prime$ が存在する． □

Theorem 2（∃-活性時間模倣関係と ∃-活性リージョン模倣関係）A_I の $inf(r_I) \cap F_I \neq \phi$ を満たすある状態列 $r_I{=}s_0', s_1', \ldots$ に対して，A_{II} の $inf(r_{II}) \cap F_{II} \neq \phi$ を満たす状態列 $r_{II}{=}s_0'', s_1'', \ldots$ が存在する．この場合に，r_I と r_{II} に関して，$RG(s_i', s_i'') \in \chi$ に対して，$R_\chi{=}\{(s_i', s_i'') | RG(s_i', s_i'') \in \chi\}$ とする．χ が A_I から A_{II} への ∃-活性リージョン模倣関係である必要十分条件は，R_χ が A_I から A_{II} への ∃-活性時間模倣関係 $A_I \preceq^{\exists} A_{II}$ である．

（証明の方針）

r_I と r_{II} に関して，Theorem 1（安全性時間模倣関係と安全性リージョン模倣関係）の証明と同様な方法を展開すればよい．紙面の都合上から，省略する． □

Definition 11（∀-活性リージョン模倣関係の定義）A_I の $inf(r_I) \cap F_I \neq \phi$ を満たすすべての状態列 $r_I{=}s_0', s_1', \ldots$ ごとに，A_{II} の $inf(r_{II}) \cap F_{II} \neq \phi$ を満たす状態列 $r_{II}{=}s_0'', s_1'', \ldots$ が存在する場合に，r_I と r_{II} に関して，$RG(s_i', s_i'') \in \chi$ に対し，以下の条件が満たされるときに限り，$\chi \subseteq RG_{A_I \| A_{II}}$ は，A_I から A_{II} への ∀-活性リージョン模倣関係であるという：

(1) 任意の $\sigma \in \Sigma$ に対して，

$$\langle s_i', [\nu_i] \rangle \xrightarrow{\sigma} \langle s_{i+1}', [\nu_{i+1}] \rangle$$

ならば，$RG(s_{i+1}', s_{i+1}'') \in \chi$ である $\langle s_{i+1}'', [\nu_{i+1}] \rangle$ に対して，

$$\langle s_i'', [\nu_i] \rangle \xrightarrow{\sigma} \langle s_{i+1}'', [\nu_{i+1}] \rangle$$

である．

(2) $\forall s_0' \in S_{0I}$ に対して，$RG(s_0', s_0'') \in \chi$ を満たす $s_0'' \in S_{0II}$ が存在する． □

Theorem 3（∀-活性時間模倣関係と ∀-活性リージョン模倣関係）A_I の $inf(r_I) \cap F_I \neq \phi$ を満たすすべての状態列 $r_I{=}s_0', s_1', \ldots$ ごとに，A_{II} の $inf(r_{II}) \cap F_{II} \neq \phi$ を満たす状態列 $r_{II}{=}s_0'', s_1'', \ldots$ が存在する．この場合に，r_I と r_{II} に関して，$RG(s_i', s_i'') \in \chi$ に対して，$R_\chi{=}\{(s_i', s_i'') | RG(s_i', s_i'') \in \chi\}$ とする．χ が A_I から A_{II} への ∀-活性リージョン模倣関係である必要十分条件は，R_χ が A_I から A_{II} への ∀-活性時間模倣関係 $A_I \preceq^{\forall} A_{II}$ である．

（証明の方針）

r_I と r_{II} に関して，Theorem 1（安全性時間模倣関係と安全性リージョン模倣関係）の証明と同様な方法を展開すればよい．紙面の都合上から，省略する． □

Example 3（リージョン模倣関係の事例）図 5 の事例により，上位レベルの仕様 A_I と下位レベルの仕様 A_{II} に対して，リージョン模倣関係と時間模倣関係を示す．

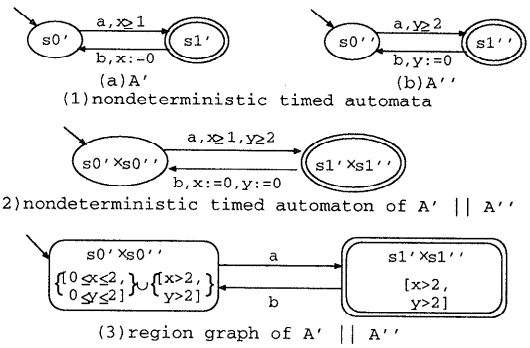

図 5 リージョン模倣関係の例

Fig. 5 Example of region simulation relation.

まず，A_I から A_{II} への安全性リージョン模倣関係と安全性時間模倣関係を示す．

(1) まず，図 5 の (1) により，A_I から A_{II} への安全性時間模倣関係を考える．

(a) $(s_0', s_0'') \in R_\chi$ ならば $a, x \geq 2 \wedge y \geq 2$ について，

$$s_0' \xrightarrow{a, x \geq 2 \wedge y \geq 2} s_1'$$

である s_1' が存在するならば，

$$s_0'' \xrightarrow{a, x \geq 2 \wedge y \geq 2} s_1''$$

である s_1'' が存在して，$(s_1', s_1'') \in R_\chi$ である．

同様に，$(s_1', s_1'') \in R_\chi$ ならば，$(s_0', s_0'') \in R_\chi$ である．

(b) $(s_0', s_0'') \in R_\chi$ を満たす s_0'' が存在する．

以上より，A_I から A_{II} への安全性時間模倣関係が存在する．

(2) 次に，図 5 の (3) により，A_I から A_{II} への安全性リージョン模倣関係を考える．

(a) $RG(s_0', s_0'') \in \chi$ に対して，

$$\langle s_0', \{[0 \leq x \leq 2, 0 \leq y \leq 2]\} \cup \{[x > 2, y > 2]\} \rangle \xrightarrow{a} \langle s_1', [x > 2, y > 2] \rangle$$

ならば，$RG(s_1', s_1'') \in \chi$ である $\langle s_1'', [x > 2, y > 2] \rangle$ に対して，

$$\langle s_0'', \{[0 \leq x \leq 2, 0 \leq y \leq 2]\} \cup \{[x > 2, y > 2]\} \rangle \xrightarrow{a} \langle s_1'', [x > 2, y > 2] \rangle$$

である．

同様に，$RG(s_1', s_1'') \in \chi$ に対して，

$RG(s_0\prime, s_0\prime\prime) \in \chi$ である．

(b) $RG(s_0\prime, s_0\prime\prime) \in \chi$ を満たす $s_0\prime\prime \in S_0\prime\prime$ が存在する．

以上より，$A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係が存在する．

次に，$A\prime$ から $A\prime\prime$ への活性リージョン模倣関係と活性時間模倣関係を示す．

(1) まず，図5の(1)により，$A\prime$ から $A\prime\prime$ への活性時間模倣関係を考える．この例では，$A\prime$ の $inf(r\prime) \cap F\prime \neq \phi$ を満たすすべての状態列 $r\prime$=$s_0\prime, s_1\prime, \ldots$ ごとに，$A\prime\prime$ の $inf(r\prime\prime) \cap F\prime\prime \neq \phi$ を満たす状態列 $r\prime\prime$=$s_0\prime\prime, s_1\prime\prime, \ldots$ が存在する．この状態列は，安全性時間模倣関係で考慮した状態列そのものである．明らかに，$r\prime$ と $r\prime\prime$ に関して，$RG(s_i\prime, s_i\prime\prime) \in \chi$ に対して，$\chi \subseteq RG_{A\prime \| A\prime\prime}$ は，$A\prime$ から $A\prime\prime$ への ∃-活性時間模倣関係および ∀-活性時間模倣関係である．

(2) 次に，図5の(3)により，$A\prime$ から $A\prime\prime$ への活性リージョン模倣関係を考える．この例では，安全性リージョン模倣関係で考慮した状態列が ∃-活性リージョン模倣関係および ∀-活性リージョン模倣関係を満たすので，$A\prime$ から $A\prime\prime$ への ∃-活性リージョン模倣関係および ∀-活性リージョン模倣関係が存在する． □

本論文では，上位レベルの仕様 $A\prime$ から下位レベルの仕様 $A\prime\prime$ への時間模倣関係が存在すれば，下位レベルの仕様 $A\prime\prime$ は上位レベルの仕様 $A\prime$ を充足すると考える．

Theorem 4（**時間模倣関係と言語包含関係の関係**）時間模倣関係は言語包含関係より強い関係である．

（証明）

定義より，時間模倣関係が成立すれば言語包含関係は明らかに成立する．しかし，言語包含関係が成立しても時間模倣関係が成立しない場合がある．**図6**に示すように，(1) 非決定性時間オートマトンと (2) 非決定性時間オートマトンは，

$$(s_0 \xrightarrow{a,x\leq 1} s_1 \xrightarrow{b,x>1} s_2 \xrightarrow{d,x:=0} s_0)^{\omega}$$

または

$$(s_0 \xrightarrow{a,x\leq 1} s_1 \xrightarrow{c,x>1} s_3 \xrightarrow{d,x:=0} s_0)^{\omega}$$

を受理するので，言語包含関係 (1) ⊆ (2) かつ (2) ⊆ (1) が成立している．しかし，(1) は $a, x \leq 1$ を受理すると $b, x > 1$ と $c, x > 1$ の両方を受理できるが，(2) は $a, x \leq 1$ を受理すると $b, x > 1$ または $c, x > 1$ の一方しか受理できない．ゆえに，言語包含関係が成立しても安全性模倣関係が成立していない．

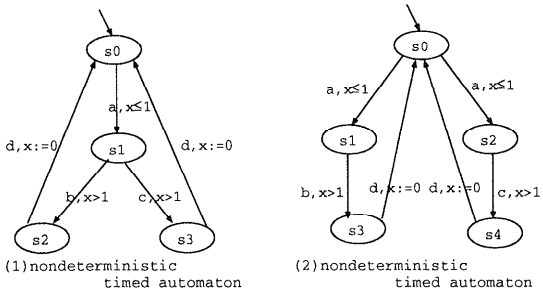

図6 安全性時間模倣関係と言語包含関係

Fig. 6 Relation between safe timed simulation and language inclusion.

他の模倣関係の場合もまったく同様である．

以上より，時間模倣関係は言語包含関係より強い関係である． □

Theorem 5（**∃-活性時間模倣と ∀-活性時間模倣の関係**）∀-活性時間模倣関係は ∃-活性時間模倣関係より強い関係である．

（証明）

まず，定義より明かに，∀-活性時間模倣関係は ∃-活性時間模倣関係より強い関係である．

次に，∃-活性時間模倣関係は満たすが，∀-活性時間模倣関係は満たさない例を**図7**に示す．すべての公平性を満たす下位レベルの仕様の状態列は

$$s_0 \xrightarrow{a,x<1} (s_1 \xrightarrow{a} s_1)^{\omega} \xrightarrow{a,x\geq 1} (s_2 \xrightarrow{b} s_2)^{*}$$

および

$$s_0 \xrightarrow{b,x<1} (s_3 \xrightarrow{a} s_3)^{\omega} \xrightarrow{b,x\geq 1} (s_4 \xrightarrow{b} s_4)^{*}$$

および

$$s_0 \xrightarrow{c,x<1} (s_5 \xrightarrow{a} s_5)^{\omega} \xrightarrow{c,x\geq 1} (s_6 \xrightarrow{b} s_6)^{*}$$

および

$$s_0 \xrightarrow{d,x<1} (s_7 \xrightarrow{a} s_7)^{\omega} \xrightarrow{d,x\geq 1} (s_8 \xrightarrow{b} s_8)^{*}$$

である．一方，公平性を満たす上位レベルの仕様の状態列

$$v_0 \xrightarrow{a,x<1} (v_1 \xrightarrow{a} v_1)^{\omega} \xrightarrow{a,x\geq 1} (v_2 \xrightarrow{b} v_2)^{*}$$

が存在する．以上より，上位レベルの仕様の状態列に対して，下位レベルの仕様の状態列が存在するので，∃-活性時間模倣関係が存在する．しかし，公平性を満たす上位レベルの仕様の状態列

$$v_0 \xrightarrow{b,x<1} (v_3 \xrightarrow{a} v_3)^{*} \xrightarrow{b,x<1} (v_4 \xrightarrow{b} v_4)^{\omega}$$

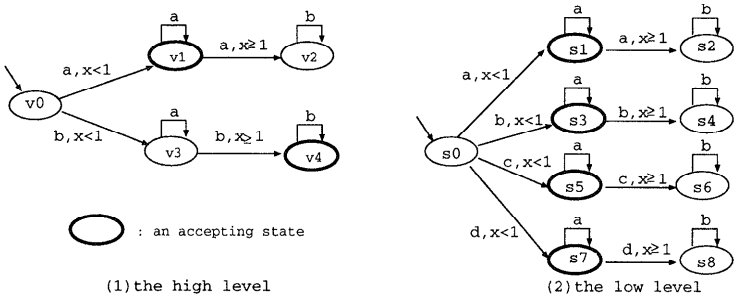

図7 ∃-活性時間模倣と∀-活性時間模倣の関係
Fig. 7 Relation between ∃-live timed simulation and ∀-live timed simulation.

に対しては，下位レベルの仕様の状態列が存在しない．以上より，上位レベルの仕様から下位レベルの仕様へ∀-活性時間模倣関係が存在しない． □

3.2.3 時間模倣関係の検証アルゴリズム

時間模倣関係の存在性は，リージョン模倣関係の存在性に帰着して，検証する．ゆえに，非決定性時間オートマトンの間の時間模倣関係の存在性の検証アルゴリズムは以下の手続きから構成される．まず，下位レベルの仕様と上位レベルの仕様の非決定性時間オートマトンのカルテジアン積を構成し，次に，カルテジアン積のリージョングラフを構成し，最後に，上位レベルの仕様から下位レベルの仕様へのリージョン模倣関係の存在性を検証する．

Definition 12（安全性時間模倣関係のアルゴリズム） 非決定性時間オートマトンの間の安全性時間模倣関係に基づく検証アルゴリズムは以下の手続きから構成する．

(1) 上位レベルの仕様と下位レベルの仕様の非決定性時間オートマトンのカルテジアン積を構成する．

(2) カルテジアン積のリージョングラフを構成する．
  (a) タイミング定数の組合せによって，クロックリージョンを構成する．
  (b) 各状態をクロックリージョンに分割して，リージョングラフの状態集合を構成する．
  (c) リージョングラフの状態集合をイベントで結ぶ．

(3) リージョングラフ上の安全性リージョン模倣関係を次のような手続きでチェックする．基本的考え方は安全性リージョン模倣関係 R を $R(0),\ldots,R(k),R(k+1)$ と帰納的に計算する．そして，$R(k)=R(k+1)$ となる $R(k)$ を $R=R(k)$ とする．なお，上位レベルの仕様 $A\prime=(\Sigma,S\prime,S_0\prime,C\prime,E\prime,F\prime)$ と下位レベルの仕様 $A\prime\prime=(\Sigma,S\prime\prime,S_0\prime\prime,C\prime\prime,E\prime\prime,F\prime\prime)$ に対して，$A\prime \parallel A\prime\prime$ のリージョングラフを $RG_{A\prime\|A\prime\prime}=(Q,Q_0,\Sigma,E,F)$ とする．ここで，

Q：$\langle s\prime\times s\prime\prime,[\nu]\rangle$ の有限集合

$s\prime\in S\prime$, $s\prime\prime\in S\prime\prime$

$[\nu]=\{\nu\prime|\nu\equiv\nu\prime\}$, $\nu\in\Gamma(A\prime\parallel A\prime\prime)$

Q_0：$\langle s_0\prime\times s_0\prime\prime,[\nu_0]\rangle$ の有限集合

$s_0\prime\in S_0\prime$, $s_0\prime\prime\in S_0\prime\prime$

$[\nu_0]$：すべてのクロック変数が 0 の順序対の集合

Σ：有限イベント集合

$E\subseteq Q\times\Sigma\times Q$：状態遷移関係

F：$\forall s\prime\in F\prime$ と $\forall s\prime\prime\in F\prime\prime$ に対する $\langle s\prime\times s\prime\prime,[\nu]\rangle$ の有限集合

以下に，$R(k)$ を計算するアルゴリズムを示す．

(a) まず，$R(0)$ に $\langle s\prime\times s\prime\prime,[\nu]\rangle$ のすべての有限集合を代入する．さらに，初期化 $k:=0$ する．

(b) 次に，以下の手続きを繰り返して，帰納的に $R(k)$ から $R(k+1)(k\geq 0)$ を計算する．
  (i) $R(k+1)=\phi$ とする．
  (ii) すべての $(s_i\prime,s_i\prime\prime)\in R(k)$ に対して，$\forall\sigma\in\Sigma$ について，
  もし，

$$\langle s_i\prime,[\nu_i]\rangle\xrightarrow{\sigma}\langle s_{i+1}\prime,[\nu_{i+1}]\rangle$$

  である $\langle s_{i+1}\prime,[\nu_{i+1}]\rangle$ が存在するならば，

$$\langle s_i\prime\prime,[\nu_i]\rangle\xrightarrow{\sigma}\langle s_{i+1}\prime\prime,[\nu_{i+1}]\rangle$$

  である $\langle s_{i+1}\prime\prime,[\nu_{i+1}]\rangle$ が存在して，$(s_{i+1}\prime,s_{i+1}\prime\prime)\prime\in R(k)$ である．上記条件を満たすときに，$R(k+1)=R(k+1)\cup\{(s_i\prime,s_i\prime\prime)\}$ とする．

(iii) もし，$R(k+1)=R(k)$ ならば，次の (c) にいく．もし，$R(k+1)\neq R(k)$ ならば，$k:=k+1$ として，(b) の (i) に戻る．

(c) $\forall s_0\prime \in S_0$ に対して，$(s_0\prime, s_0\prime\prime)\in R(k)$ を満たす $s_0\prime\prime \in S_0\prime\prime$ が存在するときかつそのときに限り，$A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係が存在すると判定する．

以上の (a)，(b)，(c) のすべての手続きは最大不動点として形式化できるので，手続きは停止する．(a)，(b)，(c) のすべての手続きにより，$A\prime$ から $A\prime\prime$ への安全性リージョン模倣関係が存在すれば，下位レベルの仕様 $A\prime\prime$ は上位レベルの仕様 $A\prime$ を充足すると判定する． □

Definition 13（∃-活性時間模倣関係のアルゴリズム）安全性時間模倣関係と同様に，∃-活性時間模倣関係の存在性を，∃-活性リージョン模倣関係の存在性に帰着して計算する．検証アルゴリズムは以下の手続きから構成する．

(1) 安全性リージョン模倣関係を充足する R を計算する．R の状態対に制限したリージョングラフ $P=(W, W_0, \Sigma, N, Acc)$ を定義する．ここで，

W：$\langle s\prime \times s\prime\prime, [\nu]\rangle$ の有限集合

$s\prime \in S\prime$, $s\prime\prime \in S\prime\prime$

$[\nu]=\{\nu\prime | \nu \equiv \nu\prime\}$, $\nu \in \Gamma(A\prime \parallel A\prime\prime)$

W_0：$\langle s_0\prime \times s_0\prime\prime, [\nu_0]\rangle$ の有限集合

$s_0\prime \in S_0\prime$, $s_0\prime\prime \in S_0\prime\prime$

$[\nu_0]$：すべてのクロック変数が 0 の順序対の集合

Σ：有限イベント集合

$N \subseteq W \times \Sigma \times W$：状態遷移関係

Acc：$\forall s\prime \in F\prime$ と $\forall s\prime\prime \in F\prime\prime$ に対する $\langle s\prime \times s\prime\prime, [\nu]\rangle$ の有限集合

(2) リージョングラフ P に関して以下の計算を行う．

(a) 初期状態ごとに，初期状態 $s_0\prime \in S_0\prime$ から到達可能な，$A\prime$ の受理状態集合 $F\prime$ の要素を含む強連結成分の集合 SCC を計算する．なお，初期状態から到達可能で $F\prime$ の要素を含む強連結成分を求めることは，$inf(r\prime)\cap F\prime \neq \phi$ を満たす $r\prime$ を求めることに対応している．

(b) 上記計算で求めたある強連結成分 $scc \in SCC$ が，$A\prime\prime$ のある受理状態集合 $F\prime\prime$ の要素を含めば，R は ∃-活性時間模倣関係であると判定する．そうでなければ，R は ∃-活性時間模倣関係でないと判定する． □

Definition 14（∀-活性時間模倣関係のアルゴリズム）安全性時間模倣関係や ∃-活性時間模倣関係と同様に，∀-活性時間模倣関係の存在性を，∀-活性リージョン模倣関係の存在性に帰着して計算する．検証アルゴリズムは以下の手続きから構成する．

(1) Definition 13（∃-活性時間模倣関係のアルゴリズム）の (1) と同一である．

(2) リージョングラフ P に関して以下の計算を行う．

(a) 初期状態ごとに，初期状態 $s_0\prime \in S_0\prime$ から到達可能な，$A\prime$ の受理状態集合 $F\prime$ の要素を含む強連結成分の集合 SCC を計算する．なお，初期状態から到達可能で $F\prime$ の要素を含む強連結成分を求めることは，$inf(r\prime)\cap F\prime \neq \phi$ を満たす $r\prime$ を求めることに対応している．

(b) 上記計算で求めたすべての強連結成分 $scc \in SCC$ ごとに，$A\prime\prime$ のある受理状態集合 $F\prime\prime$ の要素を含めば，R は ∀-活性時間模倣関係であると判定する．そうでなければ，R は ∀-活性時間模倣関係でないと判定する． □

4. 実時間システムの階層的な設計の事例

4.1 設計支援システム

本手法を支援する設計支援システムは**図 8** のようにカルテジアン積生成器，リージョングラフ生成器と時間模倣検証器から構成した．カルテジアン積生成器はプログラミング言語形式で入力した仕様から，非決定性時間オートマトンの積を生成する．リージョングラフ生成器は非決定性時間オートマトンの積からリージョングラフを自動生成する．リージョングラフは隣接リスト構造で表現して，次の時間模倣検証器に入力する．時間模倣検証器は安全性時間模倣関係および ∃-活性時間模倣関係，∀-活性時間模倣関係の 3 種類の時間模倣関係の検証を実現する．

本設計支援システムにより，上位レベルの仕様から下位レベルの仕様への時間模倣関係が存在するかどうかを検証する．もし，存在すれば，さらに下位レベルの仕様を具体化して，詳細な下位レベルの仕様を設計する．そうでなければ，下位レベルの仕様を再設計して，時間模倣関係の存在性を調べる．そして，同様に，下位レベルの仕様から詳細な下位レベルの仕様への時間模倣関係が存在するかどうかを検証する．以上の設

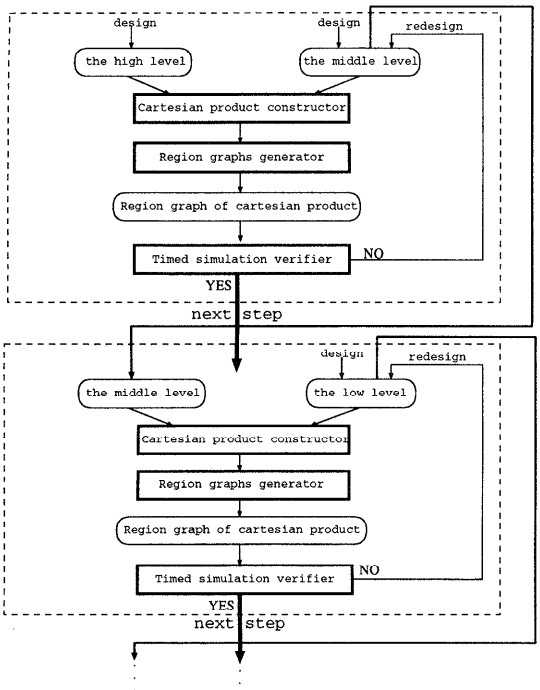

図8 設計作業の手順と設計支援システム
Fig. 8 Design processes and design support system.

計作業を続けて，最終的な実現仕様を設計する．

なお，SUN ULTRA（メインメモリ 24 MB）上にインプリメントした設計支援システムの全体規模は C 言語で約 7 Kstep である．設計作業の手順と設計支援システムを図 8 に示す．

4.2 階層的な設計の事例

本論文では，イーサネットの CSMA/CD[10] を基礎とした事例により，提案した階層的な設計手法の有効性を示す．

4.2.1 CSMA/CD の仕様記述

イーサネットの CSMA/CD は LAN で広く使われており，送信局と受信局からなり，以下に送信局と受信局の各々を詳細に説明する．

1. 送信局はデータを送信する ($send_i$) と，チャネルの応答を感知する．もし，チャネルがアイドルならば送信局はデータを送信する．しかし，チャネルが busy であったり ($busy_i$) データが破壊されたら (cd_i)，10 時刻未満待って再送する (tau_i)．このイーサネットの CSMA/CD プロトコルの送信局の仕様は以下のように記述できる．

(1) まず，**図 9** (1) のように，CSMA/CD プロトコルの i-送信局の上位レベルの仕様を設計して，非決定性時間オートマトンで仕様記述する．この仕様は，データを送信すると，チャネルの busy やデータ破壊を有限回繰り返した後，いつかは正常にデータが送信できることを意味する．すなわち，公平性の概念の導入により，チャネル上でのある有限のデータ喪失が仕様記述できる．通常の有限語上のオートマトンでは，この性質は仕様記述することが困難である．

(2) 次に，図 9 (2) のように，CSMA/CD プロトコルの i-送信局の下位レベルの仕様を設計して，非決定性時間オートマトンで仕様記述する．下位レベルの仕様では，初期状態から非決定的に動作する．一方の動作はチャネルの busy やデータ破壊を有限回繰り返した後，いつかは正常に動作して，他方の動作はネットワーク上の何らかのトラブルにより，データの送信が失敗する可能性があることを意味する．すなわち，このような異常処理が非決定性時間オートマトンの非決定性動作によって，簡単に仕様記述できる．

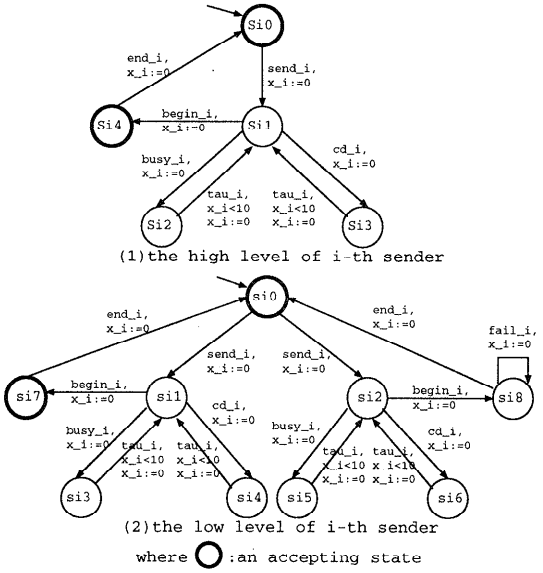

図9 CSMA/CD プロトコルの送信局の仕様記述例
Fig. 9 Example of sender of CSMA/CD protocol.

2. 受信局は 5 時刻未満でデータ受信の用意し ($ready_i$)，データが送信されると ($begin_i$)，10 時刻以内でデータを受信する (tau_i)．もし，データ受信中にチャネルが busy になったら ($busy_i$)，10 時刻以内のデータの受信待ちをして ($fail_i$)，その後データの受信を再開する．もし，データが破壊されたら，最初からデータを受信しなおす ($fail_i$)．このイーサネットの CSMA/CD プロトコルの受信局の仕様は以下のように記述できる．

(1) まず，**図 10** (1) のように，CSMA/CD プロト

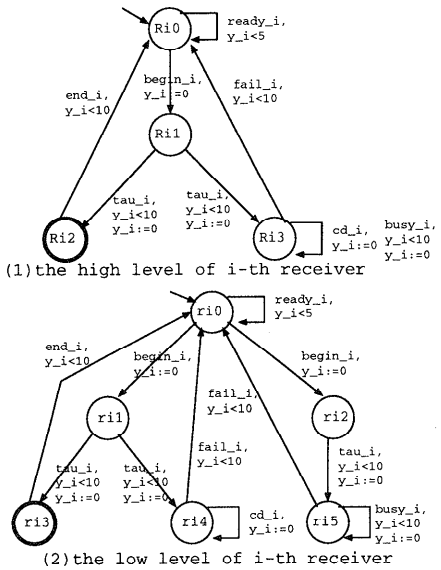

図10 CSMA/CDプロトコルの受信局の仕様記述例
Fig. 10 Example of receiver of CSMA/CD protocol.

コルの i-受信局の上位レベルの仕様を設計して，非決定性時間オートマトンで仕様記述する．この仕様は，データを受信すると，チャネルのbusyやデータ破壊を有限回繰り返した後，いつかは正常にデータが受信できることを意味する．

(2) 次に，図10(2)のように，CSMA/CDプロトコルの i-受信局の下位レベルの仕様を設計して，非決定性時間オートマトンで仕様記述する．下位レベルの仕様では，初期状態から非決定的に動作する．一方の動作はチャネルのbusyを有限回繰り返した後，データ受信を失敗し，他方の動作はネットワーク上のトラブルがまったくなく，データの受信を正常に行えたり，データ破壊によりデータ受信を失敗することを意味する．

4.2.2 階層間の整合性の検証実験

次に，時間模倣関係により，設計した上位レベルと下位レベルの仕様の整合性を検証する．本論文では，通信システムは複数の送信局と受信局の下位レベルの仕様から構成されるものと考える．すなわち，通信システムは送信局と受信局の下位レベルの仕様のカルテジアン積である．整合性検証問題では，これらの通信システムが図9または図10の上位レベルの仕様に対して，安全性時間模倣関係および∃-活性時間模倣関係，∀-活性時間模倣関係の3種類の時間模倣関係が存在するかどうかを判定する．開発した設計支援システムで整合性を検証したところ，安全性時間模倣関係および∃-活性時間模倣関係，∀-活性時間模倣関係が存在したことが分かった．本実験の事例では，公平性が重要なので，仕様間の整合性関係としては，公平性を考慮した∃-活性時間模倣関係や∀-活性時間模倣関係が適当であると考えられる．さらに，本実験の事例では，上位レベルの仕様と下位レベルの仕様との抽象度が近いので，∀-活性時間模倣関係がより適当であると考えられる．一般的に，上位レベルの仕様と下位レベルの仕様との抽象度が近い場合には，∀-活性時間模倣関係が適当であり，抽象度が遠い場合には，∃-活性時間模倣関係が適当であると考えられる．

また，送信局と受信局の総数が2個，4個，6個，8個の場合の整合性の検証実験を行って，検証の所要メモリと計算時間を，UNIXのtimeコマンドで計測した．その結果は**図11**に示すとおりである．実験結果より，送信局と受信局の総数が増加するに従って，所要メモリと計算時間が指数オーダで増加することが分かる．これは，非決定性時間オートマトンの状態数がリージョングラフの生成により，指数オーダで増加するために，検証問題がEXPTIMEの計算オーダ[11]のクラスに属することが予想される．検証コストは予想どおり大きいが，非決定性時間オートマトンの統一的な仕様記述による実時間システムの階層的設計が実現できたことは非常に重要な結果である．

5. むすび

本論文では，要求仕様から設計仕様までを，非決定性時間オートマトンで統一的に仕様記述して，公平性を含む時間模倣関係により，それらの間の整合性を自動検証できる階層的な設計手法を提案した．さらに，実時間性と公平性の概念により，既存の模倣関係を拡張および一般化して，以下の3つの時間模倣関係を提案した．

(1) 公平性を考慮しない安全性時間模倣関係
(2) ある状態列で公平性を満たす∃-活性時間模倣関係
(3) すべての状態列で公平性を満たす∀-活性時間模倣関係

今回の設計事例より，整合性としては，∃-活性時間模倣関係や∀-活性時間模倣関係が適当であることが分かった．

時間制約を考慮しないリアクティブシステムの階層的な設計を支援するシステムとしては，Cuncurrency workbench[12]やSMC[13]など多く存在する．一方，実時間システムの階層的な設計を支援するシステムとしては，Real-time Step[14]やKRONOS[15]，

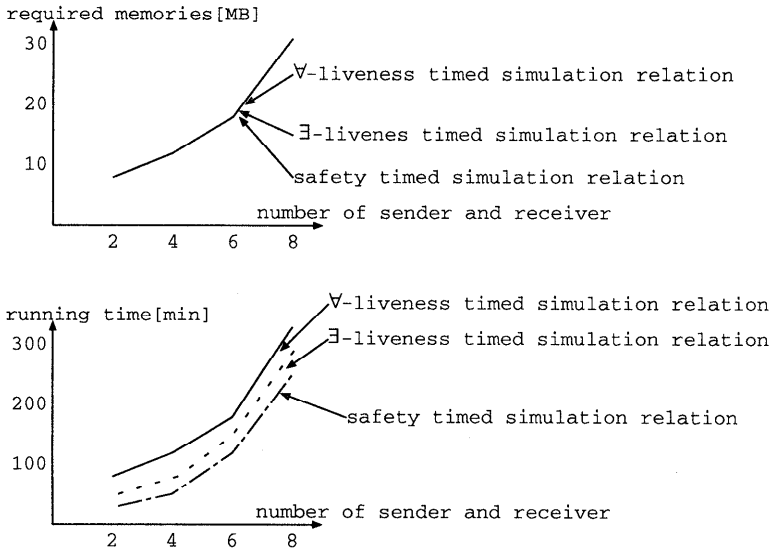

図 11 整合性検証の計算時間と所要メモリ量
Fig. 11 Running time and rerquired memories of verification.

Real-time COSPAN[16], UPPAAL[17]などが存在する．Real-time Step は実時間時相論理の証明系であり，KRONOS と UPPAAL は実時間シンボリックモデルチェッカであり，階層設計にはあまり有効ではない．また，Real-time COSPAN は抽象度の異なる仕様間に準同形写像を定義して，階層的設計を支援しているが，手作業で準同形写像を定義する必要があり，あまり実用的ではない．以上により，世界ではじめて，実時間システムの完全な自動階層設計支援システムが構築できて，その有効性が確認できた．

また，本論文で提案した時間模倣関係は，非時間模倣関係[19]と異なり，実行可能な時刻も含めたイベント間の模倣関係である．実時間システムでは，実行可能な時刻が非常に重要なので，本論文では，実行可能な時刻も考慮した時間模倣関係を採用した．さらに，時間模倣関係には，内部イベントの考慮の有無により，弱双模倣等価性と強双模倣等価性[20]の立場があり，本論文では，内部イベントを考慮しない強双模倣等価性の立場から，時間模倣関係を定義した．どちらの立場が階層的設計に適切であるかは，設計対象に依存すると考えられ，今後の評価が必要な部分である．

今後の課題としては以下が考えられる．

(1) BDDs (Binary Decision Diagrams)[18]を基礎とするシンボリック検証技術による検証の所要メモリ量の削減
(2) リージョングラフを生成しないで時間不等式手法による検証の計算時間の削減
(3) 抽象実行や構成的検証手法などによる所要メモリ量と計算時間の削減
(4) 理論的に，検証問題がどの計算クラスに属するかについての考察

参 考 文 献

1) Francez, N.: Fairness, *Texts and monographs in computer science*, p.295, Springer-Verlag (1986).
2) Thomas, W.: Automata on Infinite Objects, *Handbook of Theoretical Computer Science*, Vol.B, pp.133–191 (1990).
3) Alur, R. and Dill, D.: *The theory of Timed automata*, Lecture Notes in Computer Science, Vol.600, pp.45–73 (1992).
4) Alur, R., Courcoubetis, C. and Dill, D.: Model checking for real-time systems, *Proc. 5th LICS*, pp.414–425 (1992).
5) Milner, R.: Communication and Concurrency, p.260, Prentice Hall (1989).
6) Nakata, A., Higashino, T. and Taniguchi, K.: Timing-Action Alternating Model for Timed LOTOS and its Symbolic Verification of Bisimulation Equivalence, *Proc. Formal Description Technique IX* (1996).
7) Lynch, N.A. and Tuttle, M.R.: Hierarchical correctness proofs for distributed algorithms, *Proc. ACM Symp. on Principles of Distributed Computing*, pp.137–151 (1987).
8) Lynch, N.A. and Attiya, H.: Using mapping to prove timing properties, *Distributed Computing*, No.6, pp.121–139 (1992).
9) Dill, D., Hu, A.J. and Wong-Toi, H.: *Check-*

ing for Language Inclusion Using Simulation Preorders, Lecture Notes in Computer Science, Vol.575, pp.255–265 (1991).
10) IEEE Computer Society: IEEE ANSI/IEEE 802.3, ISO/DIS 8802/3, IEEE Computer Society Press (1985).
11) Hopcroft, J.E. and Ullman, J.D.: *Introduction to automata theory, languages, computation*, p.418, Addison-Wesley (1979).
12) Cleaveland, R., Parrow, J. and Steffen, B.: *The concurrency workbench*, Lecture Notes in Computer Science, Vol.407, pp.24–37 (1989).
13) Burch, J.R., Clarke, E.M., McMillan, K.L., Dill, D. and Hwang, L.J.: Symbolic Model Checking: 10^{20} States and Beyond, *Proc. 5th LICS*, pp.428–439 (1990).
14) Kesten, Y., Manna, Z. and Pnueli, A.: *Verifying clocked transition systems*, Lecture Notes in Computer Science, Vol.1066, pp.13–40 (1996).
15) Daws, C., Olivero, A., Tripakis, S. and Yovine, S.: *The tool KRONOS*, Lecture Notes in Computer Science, Vol.1066, pp.208–219 (1996).
16) Alur, R. and Kurshan, R.: *Timing analysis in COSPAN*, Lecture Notes in Computer Science, Vol.1066, pp.220–231 (1996).
17) Bengtsson, J., Larsen, K., Larsson, F., Pettersson, P. and Wang, Y.: *UPPAAL-a tool suite for automatic verification of real-time systems*, Lecture Notes in Computer Science, Vol.1066, pp.232–243 (1996).
18) Bryant, R.E.: Graph-based algorithms for boolean function manipulation, *IEEE Trans. Comput.*, Vol.C-35, No.8, pp.677–691, IEEE Computer Society (1986).
19) Larsen, K.G. and Yi, W.: *Time abstract bisimulation: Implicit specifications and decidability*, Lecture Notes in Computer Science, Vol.802, pp.160 176 (1994).
20) Yi, W.: *Real-time behavior of asynchronous agents*, Lecture Notes in Computer Science, Vol.458, pp.502–520 (1990).

(平成10年10月5日受付)
(平成11年 5 月 7 日採録)

山根 智（正会員）

1984年京都大学大学院修士課程修了．同年富士通（株）勤務．以後，島根大学理学部勤務．現在，鹿児島大学工学部情報工学科勤務．助教授．工学博士．ハードリアルタイムシステムや分散システム等の形式的検証の研究に従事．EATCS, ACM, IEEE等会員．