

パラメータを持つ実時間プロトコル仕様のモデル検査

中田 明夫 大阪大学大学院基礎工学研究科助手

1 まえがき

信頼性の高い通信プロトコルの設計技法の一つとしてモデル検査(有限状態機械で記述したプロトコルの動作仕様が時相論理式で記述した要求仕様を満足するか否かの検査)がある。特に近年、実時間通信プロトコル仕様のモデル検査技法は近年非常に注目を集めてきている。

従来のモデル検査技法では動作仕様及び要求仕様における設計パラメータを完全に指定した後、動作仕様が要求仕様を満たしているか否かをYes/Noで判定するものであった。しかし、設計の段階ではパラメータの値を試行錯誤で決めるのではなく、むしろ動作仕様が要求仕様を満足するようなパラメータの値に関する条件を求められる方が望ましい。最近、時間オートマトン[1]で記述された動作仕様がパラメータを含む時相論理式で記述された要求仕様を満足するためのそのパラメータに関する条件を導出する方法[2]が発表されたが、その手法は動作仕様がパラメータを持つ場合には適用できない。現実の設計過程においてはむしろ動作仕様の側にパラメータを持たせて、要求仕様を満足できるような値を探すほうが自然である。また、さらに要求仕様のパラメータも完全に固定ではなくある程度の範囲で動かしてもよいという場合もあると思われる。そのようなとき要求仕様のパラメータ群と動作仕様のパラメータ群がどのような関係式を満たせば動作仕様が要求仕様を満足するのかということが分かれば、実時間プロトコルの設計過程におけるパラメータを決定する作業の大きな助けになるとと思われる。

そこで本研究では、時相論理式で記述された実時間プロトコルの要求仕様と状態遷移モデルで記述された実時間プロトコルの動作仕様が与えられ、双方にパラメータが記述されているとき、動作仕様が要求仕様を満足するための双方のパラメータ群に関する必要十分条件を自動的に導出する方法を提案する。

本論文では動作仕様の記述モデルとしてA-TSLTS(Alternating Timed Symbolic Labelled Transition System)[3]を用いる。A-TSLTSは実時間システムを記述できるオートマトンモデルの一種である。A-TSLTSは有限個の状態を持ち、各状態はいくつかの状態変数(パラメータ)を持っている。また、各遷移は時間経過による遷移(時間遷移)または(入出力)動作による遷移(動作遷移)のいずれかである。時間遷移は経過時間を代入する変数(時間変数)および遷移条件からなり、動作遷移は動作名および遷移条件からなる。各遷移条件は時間変数およびパラメータ変数を用いた線形不等式の論理結合で記述する。各状態は時間遷移のみが可能な状態(休止状態)か動作遷移のみが可能な状態(活動状態)のいずれかであるとし、休止状態からは活動状態へ、活動状態からは休止状態に交互に遷移するとする。A-TSLTSはAlurらが提案した時間オートマトン[1]と異なり、状態 s から状態 s' へ到達するための状態 s のパラメータに関する条件が s から s' までの遷移条件の論理結合で表現できるという特徴がある。

要求仕様を記述する実時間時相論理としてはTCTL(Timed Computation Tree Logic)[4]を採用する。TCTLは例えば「ある動作 a の実行後 t 秒以内にある動作 b が実行可能にならないといけない」などといった性質(要求)を記述することができる論理である。本研究ではモデルA-TSLTSの特徴を生かし、TCTLの各構文要素 f とA-TSLTSの状態 s の組 (s, f) から s が性質 f を満たすための s の状態変数(パラメータ)に関する条件 $PT(s, f)$ を再帰的に求めるアルゴリズムを提案する。

2 A-TSLTSモデル

動作仕様の記述モデルA-TSLTS[3]は以下のように定義される。

定義2.1 $M = \langle S, Act, Var, DVar(), Pred, E, s_0 \rangle$ をA-TSLTSと定義する。ここで、 S は状態名の有限集合、 Act は動作名の集合、 Var はパラメータ変数の集合、 $DVar()$ は S から Var の部分集合への写像、 $Pred$ は Var に属する変数を自由変数にもつ線形不等式の論理結合の集合、 E は遷移関係で $S \times (Var / Act) \times Pred \times S$ の部分集合、 s_0 / S は初期状態である。 d / Var に対して $(s, d, P, s') / E$ を時間遷移と呼び $s - e(d)[P] \rightarrow s'$ と記述する。 a / Act に対して $(s, a, P, s') / E$ を動作遷移と呼び $s - a[P]s'$ と記述する。A-TSLTSの各状態は休止状態(idle state)または活動状態(active state)のいずれか一方に属す。休止状態では時間遷移のみが実行可能で、活動状態に遷移する。活動状態では動作遷移のみが実行可能で、休止状態に遷移す

る。また、各状態にはパラメータ変数の集合 $DVar(s)$ が関連づけられている。以後、例えば $DVar(s) = \{x, y\}$ であるとき状態 s を $s\{x, y\}$ と書くことにする。 $s - (d)[P] \rightarrow s'$ は状態 s からある時間だけ経過したのち、状態 s' に遷移し、その経過時間を変数 d に代入するという遷移を表す。 P は遷移条件で、 d および s のパラメータ変数 $DVar(s)$ が満たすべき条件を線形不等式の論理結合で記述する。また、 $s' - a[P] \rightarrow s_1$ は状態 s' のパラメータが遷移条件 P を満たすときに動作 a を実行可能で、状態 s_1 に遷移することを表す。休止状態からは時間遷移は一本のみ記述できるとする。活動状態からは複数の動作遷移で分岐することを許す。

例2.1 図1にA-TSLTSの例を示す。図1において、遷移 $s\{x\} - e(d)[d \leq x] \rightarrow s'\{x, d\}$ は、 s から x 以下の時間だけ経過して s' に遷移し、その経過時間を変数 d に代入されることを表す。また、遷移 $s'\{x, d\} - a[d \leq x - 2] \rightarrow s_1\{x, d\}$ は s' のパラメータ d および x が $d \leq x - 2$ を満たすとき、すなわち状態 s から $x-2$ 単位時間以内に a を実行可能であることを表す。遷移 $s'\{x, d\} - b[d > x - 3] \rightarrow s_2\{x, d\}$ も同様である。

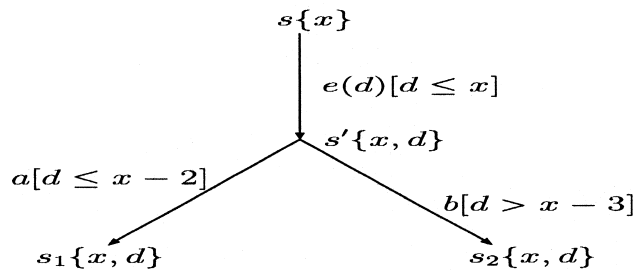


図1 ATSLTSの例

A-TSLTSで記述された動作仕様の具体的な動きは、初期状態に対してパラメータの値を具体的に決めたときに初めて定まる。例えば、例2.1の状態 s を初期状態としたとき、この動作仕様は $x=3.5$ のとき、 $s(x=3.5) - 1.5 \rightarrow s'(x=3.5, d=1.5)$ という動きが可能である。

上の例では、 $x=3.5$ という値の割り当てにおいてある d の値1.5が存在し、遷移条件 $d \leq x$ を満たすので、1.5という値の時間遷移を実行できて、状態 s' のパラメータ x, d をそれぞれ3.5, 1.5に割り当てた状態に遷移する。このような各パラメータ変数への値の割り当てを付値とよび ρ, ρ' などで表す。また、 x に値 v を割り当て、それ以外の変数には ρ と同じ値を割り当てる付値を $\rho[x=v]$ と記述する。一階述語 P の各自由変数に付値 ρ の値を割り当てたときに真となることを $\rho \models P$ と記述する。 s のパラメータの値を付値 ρ で決めたときの状態を状態 s と ρ の組 (s, ρ) で表し、 s の ρ による具体的状態と呼ぶ。

具体的状態の遷移関係の形式的な定義は省略する。詳細は文献[5]を参照されたい。

例2.2 図2にマウスなどのボタンを押す・離すといったイベントを入力として、その押し方のタイミングを判定して出力するシステム(以下、クリック判定システムと呼ぶ)の動作仕様を示す。図では中間状態のパラメータ集合は省略した。この動作仕様は設計パラメータ(=初期状態のパラメータ) $p_1, \dots, p_7$ を持つ。動作の概略は次の通りである。まず状態 s_0 で任意の時間遷移後にボタンを押す入力動作 $press$ を行うことができる。その後、時間 p_1 まで時間経過が可能で、時間経過 p_2 までにボタンを離す入力動作 $release$ か、時間経過 p_2 以上で出力動作 $longsingleclick$ を実行できる。もし $longsingleclick$ を実行すれば、 $release$ を任意のタイミングで実行し停止する。状態 s_1 で $release$ を実行後、状態 s_2 で時間 p_3 まで時間経過が可能で、 p_4 までに2回目の $press$ が実行されるか、さもなければ時間経過 p_4 以上で $shortsingleclick$ を実行し停止する。状態 s_2 で2回目の $press$ が実行されたら、状態 s_3 で時間 p_5 まで時間経過が可能で、時間 p_6 までに2回目の $release$ を実行するか、さもなければ時間 p_6 以上で $longaftershortclick$ を実行し、 $release$ を任意のタイミングで実行し停止する。状態 s_3 で2回目の $release$ が実行されれば、時間 p_7 までに動作 $doubleclick$ を実行し停止する。

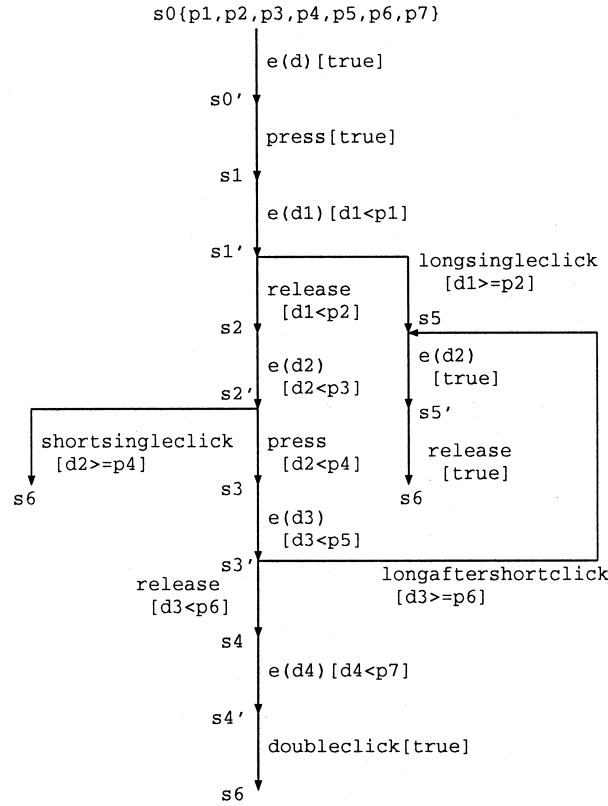


図2 クリック判定システム動作仕様

3 TCTL

本章では本研究で要求仕様を記述するTCTL(Timed Computation Tree Logic)について述べる。

定義3.1 TCTL式 f の構文は図3に示すBNFで定義される。ただし、 a は動作名、 c は非負実数定数、 $\sim$ は比較演算子 $<, /<, _, >, \geq, =$ のいずれかである。 $\sim c$ は省略可能とし、その場合は ≥ 0 が指定されたものとする。/

$f ::= true$ (恒真)	
$false$ (恒偽)	$fEU_{\sim c}f$ (存在 until 演算子)
$\neg f$ (否定)	$fAU_{\sim c}f$ (全称 until 演算子)
$f \wedge f$ (論理積)	$EG_{\sim c}f$ (存在 always 演算子)
$f \vee f$ (論理和)	$AG_{\sim c}f$ (全称 always 演算子)
$f \Rightarrow f$ (含意)	$EF_{\sim c}f$ (存在 eventually 演算子)
$\langle a \rangle_{\sim c}f$ (存在 next 演算子)	$AF_{\sim c}f$ (全称 eventually 演算子)
$[a]_{\sim c}f$ (全称 next 演算子)	

図3 TCTLの構文

TCTL式は具体的状態、すなわち、状態とパラメータへの付値の対に対してその状態からの動作系列に対して成り立って欲しい性質を記述する論理である。直観的な意味は次の通りである。 $true$ は任意の具体的状態に対して成り立つ式である。 $\neg f$ は f が成り立たない具体的状態に対して成り立つ式である。 $false$ は任意の具体的状態に対して成り立たない式であり、 $\neg true$ と同じ意味である。 f_1 / f_2 は f_1 と f_2 が共に成り立つ具体的状態で成り立つ式である。 f_1 / f_2 は $\neg((\neg f_1) / ((\neg f_2)))$ と同じ意味、 $f_1 f_2$ は $\neg(\neg(f_1 / (\neg f_2)))$ と同じ意味である。 $\langle a \rangle_{\sim c} f$ は c 単位時間以内に動作 a を実行可能で、次の状態で f が成り立つような具体的状態で成り立つ式である。 $\sim$ が $\geq, <, >, =$ の場合も同様に定義されるので、以下、 $\sim$ が $\leq$ の場合について

てのみ述べる。 $[a]_{\leq c} f$ は c 単位時間以内に動作 a を実行したならば必ず次の状態で f が成り立つような具体的状態で成り立つ式で、TCTL式 $\neg \langle a \rangle_{\leq c} \neg f$ と同じ意味である。 $f_1 EU_{\leq c} f_2$ は f_2 が成り立つ状態へ到達するある動作系列が存在して、その状態へ到達するまでの任意の状態では f_1 が成り立っているような具体的状態で成り立つ式である。 $f_1 AU_{\leq c} f_2$ は f_2 が成り立つ状態へ到達する任意の動作系列に対して、その状態へ到達するまでの任意の状態では f_1 が成り立っているような具体的状態で成り立つ式である。 $EF_{\leq c} f$ は $EU_{\leq c} f$ と同じ意味で、 c 単位時間以内に f が成り立つ状態に到達できるある動作系列が存在するような具体的状態で成り立つ式である。同様に $AG_{\leq c} f$ は $\neg EF_{\leq c} \neg f$ と同じ意味、 $AF_{\leq c} f$ は $true AU_{\leq c} f$ と同じ意味、 $EG_{\leq c} f$ は $\neg AF_{\leq c} \neg f$ と同じ意味である。

一般に、 $(s, \rho) \models f$ で TCTL式 f が A-TSLTSの具体的状態 (s, ρ) において真となることを表す。

関係 $\models$ の形式的な定義を以下に示す。なお、以下の定義では $true, \neg f, f_1 / f_2, \neg \langle a \rangle_{\leq c} f, f_1 EU_{\leq c} f_2, f_1 AU_{\leq c} f_2$ の 6 つの構文の意味定義のみ示す。他の構文はすべてこれら 6 種類の構文のみを用いて記述可能である。

定義3.2 A-TSLTSの任意の具体的状態 (s, ρ) と TCTL式 f の各構文に対して関係 $(s, \rho) \models f$ を図4のように定義する。

$$\begin{aligned}
(s, \rho) &\models true. \\
(s, \rho) \models \neg f &\stackrel{\text{def}}{=} (s, \rho) \not\models f. \\
(s, \rho) \models f_1 \wedge f_2 &\stackrel{\text{def}}{=} (s, \rho) \models f_1 \text{ かつ } (s, \rho) \models f_2. \\
(s, \rho) \models \langle a \rangle_{\sim c} f &\stackrel{\text{def}}{=} (s, \rho) \xrightarrow{a} (s', \rho') \xrightarrow{a} (s'', \rho'') \text{ なるある遷移系列が存在して} \\
&\quad t \sim c \text{ かつ } (s'', \rho'') \models f. \\
(s, \rho) \models f_1 EU_{\sim c} f_2 &\stackrel{\text{def}}{=} (s, \rho) = (s_1, \rho_1) \xrightarrow{t_1} (s'_1, \rho'_1) \xrightarrow{a_1} \cdots \xrightarrow{t_{k-1}} (s'_{k-1}, \rho'_{k-1}) \xrightarrow{a_{k-1}} (s_k, \rho_k) \\
&\quad \text{なるある遷移系列が存在して、} (s_k, \rho_k) \models f_2 \text{ かつ } t_1 + \cdots + t_{k-1} \sim c \\
&\quad \text{かつ任意の } i (1 \leq i \leq k-1) \text{ に対して } (s_i, \rho_i) \models f_1. \\
(s, \rho) \models f_1 AU_{\sim c} f_2 &\stackrel{\text{def}}{=} (s, \rho) = (s_1, \rho_1) \xrightarrow{t_1} (s'_1, \rho'_1) \xrightarrow{a_1} \cdots \xrightarrow{t_{k-1}} (s'_{k-1}, \rho'_{k-1}) \xrightarrow{a_{k-1}} (s_k, \rho_k) \\
&\quad \text{なる任意の遷移系列に対して、} (s_k, \rho_k) \models f_2 \text{ かつ } t_1 + \cdots + t_{k-1} \sim c \\
&\quad \text{かつ任意の } i (1 \leq i \leq k-1) \text{ に対して } (s_i, \rho_i) \models f_1.
\end{aligned}$$

図4 TCTL式の意味定義

例3.1 例2.2のシステム(の初期状態 s_0)に対する要求仕様の一部をTCTLで記述すると例えば図5のようになる。

- (1) $[press][release]_{< q_1} \langle shortsingleclick \rangle_{\geq q_2} true$
- (2) $[press][release]_{\geq q_1}$

図5 クリック判定システムの要求仕様記述

条件(1)はpressの後 q_1 単位時間未満でreleaseがあり、その後 q_2 単位時間以上経過したら「ショートクリック 1 回」と判定し、動作「shortsingleclick」を実行することが可能であり、 q_2 単位時間以内では決して「shortsingleclick」を実行することがないことを表す。条件(2)はpressの後 q_1 単位時間以上でreleaseがあり、その後 q_2 単位時間以上経過後に必ず動作「longsingleclick」を実行可能にする状態に到達できて、かつ、そこに到達するまでの任意の状態では「longaftershortclick」を実行できる可能性を維持することを表す。なお、 q_1, q_2 は要求仕様のパラメータである。/

4 パラメータ条件導出

本章ではループを含まないA-TSLTS仕様の状態(正確には休止状態) s およびTCTL式 f から、 s が f を満たすための s のパラメータに関する条件式を実数上のプレスブルガー算術式(以下、実数プレスブルガー式と表記)で出力するアルゴリズムを提案する。その前にまず、本研究におけるパラメータ導出問題の正確な定義を以下に示す。

定義4.1 パラメータ条件導出問題とは、A-TSLTSモデル M とその任意の休止状態 s およびTCTL式 f が与えられたとき、任意の付値 ρ に対して条件「 $\rho \models P \quad (s/\rho) \models f$ 」を満足するような実数ブレスブルガー式 P を導出する問題である。

パラメータ条件 P を具体的に求める関数 $PT(s, f)$ を以下に形式的に定義する。

定義4.2 ループを持たないA-TSLTSモデルの休止状態 s およびTCTL式 f に対して実数ブレスブルガー式を返す関数 $PT(s, f)$ を図6のように定義する。

$$\begin{aligned}
PT(s, true) &\stackrel{\text{def}}{=} true \\
PT(s, \neg f) &\stackrel{\text{def}}{=} \neg PT(s, f) \\
PT(s, f_1 \wedge f_2) &\stackrel{\text{def}}{=} PT(s, f_1) \wedge PT(s, f_2) \\
I(a, s) = \{i | s \xrightarrow{a} s' \wedge [P_i] \rightarrow s'_i \wedge [P_i] \rightarrow s_i\} \text{ のとき、} \\
PT(s, \langle a \rangle_{\sim c} f) &\stackrel{\text{def}}{=} \exists d_s [[0 \leq d_s] \wedge [d_s \sim c] \wedge P_s \wedge \bigvee_{i \in I(a, s)} [P_i \wedge PT(s_i, f)]] \\
I(s) = \{i | s \xrightarrow{a} s' \wedge [P_i] \rightarrow s'_i \wedge [P_i] \rightarrow s_i\} \text{ のとき、} \\
PT(s, f_1 EU_{\sim c} f_2) &\stackrel{\text{def}}{=} [0 \sim c \wedge PT(s, f_2)] \vee \\
&\quad [PT(s, f_1) \wedge \exists d_s [P_s \wedge \bigvee_{i \in I(s)} [P_i \wedge PT(s_i, f_1 EU_{\sim(c-d_s)} f_2)]]] \\
PT(s, f_1 AU_{\sim c} f_2) &\stackrel{\text{def}}{=} [0 \sim c \Rightarrow PT(s, f_2)] \vee \\
&\quad [PT(s, f_1) \wedge \forall d_s [P_s \Rightarrow \bigwedge_{i \in I(s)} [P_i \Rightarrow PT(s_i, f_1 AU_{\sim(c-d_s)} f_2)]]]
\end{aligned}$$

図6 アルゴリズム $\Delta PT(s, f)$

ループを持たないA-TSLTS仕様に対しては、関数 $PT(s, f)$ の再帰の停止性は明らかである。よって、以下の定理を得る。

定理4.1 ループを持たないA-TSLTS仕様 M に対して、アルゴリズム $\Delta PT(s, f)$ はパラメータ条件導出問題の解を与える。すなわち、

$$\rho[\rho \models PT(s, f) \quad (s, \rho) \models f]$$

定理4.1の証明は紙面の都合で省略する。詳細は文献[5]を参照されたい。

5 処理系による導出結果

アルゴリズム $\Delta PT(s, f)$ によってパラメータ条件を導出する処理系を作成し、例2.2のモデルおよび例3.1の要求仕様(1)と(2)の論理積に対して導出を行ったその出力結果を論理式簡約化ツールOmega¹を用いて簡約化した結果、以下の式が得られた²。

$$(q_2 \leq p_4 \quad p_5 \leq q_1) \quad (p_6 \leq q_1 < p_5 \quad q_2 \leq p_4 < p_3)$$

条件自動生成に要した時間は0.34秒、Omegaによる簡約化に要した時間は0.13秒であった。計測した計算機環境は、CPU:PentiumIII 750Mhz, Memory: 256MB, OS: Linux2.2.13である。

¹ Omegaは <http://www.cs.umd.edu/projects/omega> で得られるフリーソフトである。

² この例では各パラメータはすべて正の数であることを仮定しているので、 $0 < p_1$ などの自明な条件はすべて省略した。

6 あとがき

本論文では、A-TSLTSで記述されたループのない動作仕様およびTCTLで記述された要求仕様から、動作仕様が要求仕様を満足するために必要十分なパラメータ値に対する条件式を自動導出する一手法を提案した。また、処理系を作成してクリック判定の例題に適用し、実用的な時間で解を得ることができた。残念ながら、より大きな例に対しては式の長さが非常に長くなり束縛変数の数が増えるため、簡約化ツールの制限により簡約化ができなかった。しかし、式の生成過程で部分式毎に簡約化すればこの制限を回避することが可能になると思われる。

また、一般にA-TSLTSがループを含む場合、関数 $PT(s, f_1 EU_{-c} f_2)$ の再帰が停止しないため、そのままでは本手法は適用できない。しかし、ツリー状の状態遷移図で、末端の状態からは単純に初期状態にもどるだけという動作仕様に対しては、初期状態に戻る遷移を除去したツリーに本手法を適用することによりパラメータ条件導出が可能である。

式を生成過程で簡約化するように処理系を改良すること、および、一般のループを含む仕様に適用できるように本手法を拡張することは、今後の課題である。

参考文献

- [1] Alur, R. and Dill, :Automata For Modelling Real-Time Systems, in Paterson, M. S. ed., *Proc. of ICALP'90, Vol. 443 of Lecture Notes in Computer Science*, pp. 322-335, Springer-Verlag(1990).
- [2] Wang, F.: Parametric Timing Analysis for Modelling Real-Time Systems, *Information and Computation*, Vol. 130, No. 2, pp. 131-150 (1996)
- [3] Nakata, A., Higashino, T. and Taniguchi, K.: Time-Action Alternating Model for Timed LOTOS and its Sympolic Verification of Bisimulation Equivalence, in *Proc. of FORTE/PSTV'96*, pp.279-294, IFIP, Chapman & Hall (1996).
- [4] Alur, R., Courcoubetis, C. and Dill, D.: Model-Checking in Dense Real-time, *Information and Computation*, Vol. 104, pp. 2-34 (1993).
- [5] 新子浩、中田明夫、大場充：時相論理式を満足する実時間プロトコル仕様のパラメータ条件導出, 情処研報2000-DPS-97-9, 情報処理学会(2000).

< 発 表 資 料 >

題 名	掲載誌・学会名等	発表年月
実時間システムの動作仕様と要求仕様からのパラメータ条件導出	平成11年度電気・情報関連学会中国支部 連合大会論文集 pp. 239-240	1999年10月
時相論理式を満足する実時間プロトコル仕様の パラメータ条件導出	第97回情報処理学会マルチメディア通信 と分散処理研究会研究報告 2000-DPS-97-9	2000年3月