マルチメディア通信と分散処理 97－9
コンピュータセキュリティ 8－9
（2000．3．21）

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

新子 浩康 中田 明夫 大場 充
広島市立大学情報科学部情報数理学科

実時間プロトコルのパラメータを含んだ動作仕様およびその要求仕様（動作仕様が満たすべき性質）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する。動作仕様はA-TSLTSという状態遷移モデルで記述する。このモデルはAlurらが提案した時間オートマトンモデルと異なり、ある状態からある状態への到達可能であるためのパラメータに関する条件を遷移条件の論理積で表現できるという特徴を持つ。要求仕様は時相論理式の一種であるTCTLで記述する。TCTLによってある時間以内にある動作ができる状態に到達可能であるなどといった要求を記述できる。本手法では、ループを持たない動作仕様の任意の状態 s とTCTLの式 f の組から、s が f を満たすために必要十分な s のパラメータに関する条件式を自動導出する。

Deriving a Parameter Condition for a Behavioral Real-Time Protocol Specification Satisfying a Temporal Logic Specification.

Hiroyasu Atarashi Akio Nakata Mitsuru Ohba
Dept. of Computer Science, Faculty of Information Sciences, Hiroshima City University

In this paper, we propose a method for deriving the weakest condition for parameters of a behavioral real-time protocol specification (written in a state transition model) to satisfy a required properties written in a temporal logic formula. The behavioral specification is written in A-TSLTS, a kind of state transition models proposed in [1]. Unlike Timed Automata[4], in A-TSLTS the reachable condition for parameter variables can be easily derived in logical combination of transition conditions. The required property is written in TCTL(Timed Computation Tree Logic[2]). In TCTL we can describe properties such as, for example, from some state of the system, some action must be executed within x units of time before some other action has executed. TCTL can also describe that some property will eventually hold within y units of time and until then, some other property always holds. The TCTL formulas, as well as A-TSLTS specifications, may also contain parameter variables. The proposed method derives automatically from the given A-TSLTS state s and the TCTL formula f the weakest parameter condition such that s satisfy the property f.

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[4]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[3]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）[1]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 [4] 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)[2] 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ければならない」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。

本稿の構成は以下の通りである。まず、2 章では動作仕様の記述モデル A-TSLTS について説明する。3 章では要求仕様の記述言語である TCTL について説明する。4 章にて動作仕様の状態 s が TCTL の式 f を満足するためのパラメータに関する条件 $PT(s, f)$ を求めるアルゴリズムを提案する。5 章で結論と今後の課題を述べる。

2 A-TSLTS モデル

動作仕様の記述モデル A-TSLTS[1] 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 $S \times (Var \cup Act) \times Pred \times S$ の部分集合、$s_0 \in S$ は初期状態である。$d \in Var$ に対して $(s, d, P, s') \in E$ を**時間遷移**と呼び $s{-}e(d)[P]{\rightarrow}s'$ と記述する。$a \in Act$ に対して $(s, a, P, s') \in E$ を**動作遷移**と呼び $s{-}a[P]{\rightarrow}s'$ 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。$s{-}e(d)[P]{\rightarrow}s'$ 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。 □

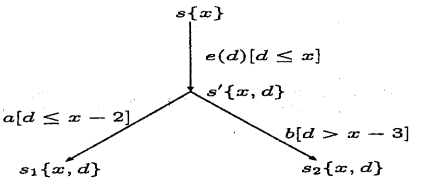

図 1: A-TSLTS の例

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、その経過時間が変数 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。 □

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。A-TSLTS 記述の具体的な動きは具体的状態の間の遷移関係として以下のように定義される。

定義 2.2 A-TSLTSM の各時間遷移 $s{-}e(d)[P]{\rightarrow}s'$ に対して具体的状態間の遷移関係を以下のように定める。

- $\rho \models \exists d[0 \leq d \wedge P]$ であるような任意の ρ および $\rho[d = t] \models [0 \leq d \wedge P]$ であるような任意

の t に対して $(s,\rho)-t\rightarrow(s',\rho[d=t])$. さらに任意の t' $(0 \leq t' \leq t)$ に対して $(s,\rho)-t'\rightarrow(s',\rho[d=t'])-(t-t')\rightarrow(s',\rho[d=t])$. [時間の連続性より]

また、各動作遷移 $s-a[P]\rightarrow s'$ に対して具体的状態間の遷移関係を以下のように定める。

- $\rho \models P$ であるような任意の ρ に対して $(s,\rho)-a\rightarrow(s',\rho)$

このように定義された具体的状態における状態遷移システムを、M の**具体的遷移システム**と呼ぶ。 □

例 2.2 $\rho[x=3.5] \models \exists d[d \leq x]$ である。また、$\rho[x=3.5, d=1.5] \models [d \leq x]$ である。よって、A-TSLTS の遷移 $s\{x\}-e(d)[d \leq x]\rightarrow s'\{x,d\}$ に対して具体的遷移 $(s,\rho[x=3.5])-1.5\rightarrow(s',\rho[x=3.5, d=1.5])$ が定まる。また、$\rho[x=3.5, d=1.5] \models [d \leq x-2]$ であるから、A-TSLTS の遷移 $s'\{x,d\}-a[d \leq x-2]\rightarrow s''\{x,d\}$ に対して具体的遷移 $(s',\rho[x=3.5, d=1.5])-a\rightarrow(s'',\rho[x=3.5, d=1.5])$ が定まる。つまり、具体的状態 (s,ρ) から 1.5 単位時間後に動作 a が実行可能である。 □

例 2.3 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)$p_1,\ldots,p_7$ 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。 □

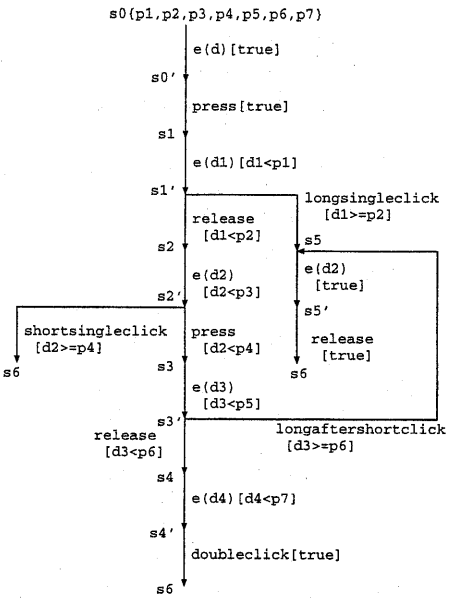

図 2: クリック判定システム動作仕様

$$
\begin{array}{rcll}
f & ::= & true & \text{(恒真)} \\
 & | & false & \text{(恒偽)} \\
 & | & \neg f & \text{(否定)} \\
 & | & f \wedge f & \text{(論理積)} \\
 & | & f \vee f & \text{(論理和)} \\
 & | & f \Rightarrow f & \text{(含意)} \\
 & | & \langle a\rangle_{\sim c} f & \text{(存在 next 演算子)} \\
 & | & [a]_{\sim c} f & \text{(全称 next 演算子)} \\
 & | & f E U_{\sim c} f & \text{(存在 until 演算子)} \\
 & | & f A U_{\sim c} f & \text{(全称 until 演算子)}
\end{array}
$$

図 3: TCTL の構文

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 $<,\leq,>,\geq,=$ 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。 □

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。$f_1 \wedge f_2$ 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。$f_1 \vee f_2$ は f_1 と f_2 の少なくともいずれか一方が成り立つ具体的状態で成り立つ式であり、$\neg((\neg f_1)\wedge(\neg f_2))$ と同じ意味である。$f_1 \Rightarrow f_2$ は f_1 が成り立たないか

$$
\begin{aligned}
(s,\rho) &\models true. \\
(s,\rho) \models \neg f &\stackrel{\text{def}}{=} (s,\rho) \not\models f. \\
(s,\rho) \models f_1 \wedge f_2 &\stackrel{\text{def}}{=} (s,\rho) \models f_1 \text{かつ } (s,\rho) \models f_2.
\end{aligned}
$$

$$
\begin{aligned}
(s,\rho) \models \langle a \rangle_{\sim c} f \stackrel{\text{def}}{=}\ & (s,\rho) - t \rightarrow (s',\rho') - a \rightarrow (s'',\rho'') \\
& \text{なるある遷移系列が存在して} \\
& t \sim c \text{ かつ } (s'',\rho'') \models f.
\end{aligned}
$$

$$
\begin{aligned}
(s,\rho) \models f_1 EU_{\sim c} f_2 \stackrel{\text{def}}{=}\ & (s,\rho) = (s_1,\rho_1) - t_1 \rightarrow (s'_1,\rho'_1) - a_1 \rightarrow \\
& \cdots - t_{k-1} \rightarrow (s'_{k-1},\rho'_{k-1}) - a_{k-1} \rightarrow (s_k,\rho_k) \\
& \text{なるある遷移系列が存在して、} \\
& (s_k,\rho_k) \models f_2 \text{かつ } t_1 + \cdots + t_{k-1} \sim c \\
& \text{かつ任意の } i(1 \leq i \leq k-1) \text{ に対して } (s_i,\rho_i) \models f_1.
\end{aligned}
$$

図 4: TCTL 式の意味定義

f_2 が成り立つかどちらかの具体的状態で成り立つ式であり、$\neg(f_1 \wedge (\neg f_2))$ と同じ意味である。$\langle a \rangle_{\leq c} f$ 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り、$\neg(\neg f_2 EU_{\leq c}(\neg f_1 \wedge \neg f_2))$ 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 $true$, $\neg f$, $f_1 \wedge f_2$, $\langle a \rangle_{\sim c} f$, $f_1 EU_{\sim c} f_2$ の 5 つの構文の意味定義のみ示す。他の構文はすべてこれら 5 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。 □

例 3.1 例 2.3 のシステム (の初期状態 s_0) に対する要求仕様を TCTL で記述すると例えば図 5 のようになる。図 5 において、例えば、条件 (1) は press の後 q_1 単位時間以内に 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 を実行し、それまでは、二回目の動作 press を実行可能な状態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。

(1) $[press][release]_{<q_1}$
$((\langle press \rangle true) EU_{\geq q_2}(\langle shortsingleclick \rangle true))$
(2) $[press][release]_{\geq q_1}$
$((\langle press \rangle true) EU_{\geq q_2}(\langle longsingleclick \rangle true))$
(3) $[press][release]_{<q_1}$
$[press]_{<q_2}[release]_{<q_1}(\langle doubleclick \rangle true)$
(4) $[press][release]_{<q_1}$
$[press]_{<q_2}[release]_{\geq q_1}(\langle longaftershortclick \rangle true)$

図 5: クリック判定システムの要求仕様記述

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 を満足するための s のパラメータに関する条件式を実数プレスブルガー文の形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「$\rho \models P \Leftrightarrow (s,\rho) \models f$」を満足するような実数プレスブルガー式 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 に対して実数プ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。ただし、図 6 において $PT'(s,f,d)$ は $PT(s,f)$ の先頭の存在量化子 $\exists d$ を除去した論理式とし、$P\{e/x\}$ は論理式 P に含まれる自由変数 x を式 e で置き換えた論理式を表すものとする。 □

関数 $PT(s,f)$ の直観的な説明は以下の通りである。まず定義 3.2 より任意の状態 s および付値 ρ に対して $(s,\rho) \models true$ であるから、明らかに $PT(s,true) = true$ となる。同様に定義 3.2 より $PT(s,\neg f) = \neg PT(s,f)$,$PT(s,f_1 \wedge f_2) = PT(s,f_1) \wedge PT(s,f_2)$ となる。$PT(s,\langle a \rangle_{\sim c} f)$ は次のようにして得られる。まず、定義 3.2 より、$(s,\rho) \models \langle a \rangle_{\sim c} f$ を満たす付値を ρ としたとき、A-TSLTS のある遷移系列 $s - e(d_s)[P_s] \rightarrow s' - a[P_i] \rightarrow s_i$ およびある時間値 (非負実数)t が存在して $(s,\rho) - t \rightarrow (s', \rho[d_s = t]) - a \rightarrow (s_i, \rho[d_s = t])$ かつ $t \sim c$ かつ $(s_i, \rho[d_s = t]) \models f$ となるはずである。$(s,\rho) - t \rightarrow (s', \rho[d_s = t])$ より $\rho[d_s = t]$ は時間遷移の遷移条件 P_s を満足する。すなわち、$\rho[d_s = t] \models P_s$ である。同様に $(s', \rho[d_s = t]) - a \rightarrow (s_i, \rho[d_s = t])$ より $\rho[d_s = t] \models P_i$ となる。また、$(s_i, \rho[d_s = t]) \models f$ より $\rho[d_s = t] \models PT(s_i, f)$ である。したがって、ρ が満たすべき条件式は

$$
\rho \models \exists d_s [0 \leq d_s \wedge d_s \sim c \wedge P_s \wedge \bigvee_{i \in I} [P_i \wedge PT(s_i, f)]]
$$

$$PT(s,true) \stackrel{\text{def}}{=} true \qquad PT'(s,true,d_s) \stackrel{\text{def}}{=} true$$

$$PT(s,\neg f) \stackrel{\text{def}}{=} \neg PT(s,f) \qquad PT'(s,\neg f,d_s) \stackrel{\text{def}}{=} \neg PT'(s,f,d_s)$$

$$PT(s,f_1 \wedge f_2) \stackrel{\text{def}}{=} PT(s,f_1) \wedge PT(s,f_2) \qquad PT'(s,f_1 \wedge f_2,d_s) \stackrel{\text{def}}{=} PT'(s,f_1,d_s) \wedge PT'(s,f_2,d_s)$$

$I(a,s) = \{i | s\text{-}e(d_s)[P_s]\rightarrow s'\text{-}a[P_i]\rightarrow s_i\}$ のとき、

$$PT(s,\langle a\rangle_{\sim c} f) \stackrel{\text{def}}{=} \exists d_s[[0 \le d_s] \wedge [d_s \sim c] \wedge P_s \wedge \bigvee_{i\in I(a,s)} [P_i \wedge PT(s_i,f)]]$$

$$PT'(s,\langle a\rangle_{\sim c} f,d_s) \stackrel{\text{def}}{=} [[0 \le d_s] \wedge [d_s \sim c] \wedge P_s \wedge \bigvee_{i\in I(a,s)} [P_i \wedge PT(s_i,f)]]$$

$I(s) = \{i | s\text{-}e(d_s)[P_s]\rightarrow s'\text{-}a_i[P_i]\rightarrow s_i\}$ のとき、

$$PT(s,f_1 EU_{\sim c} f_2) \stackrel{\text{def}}{=} \exists d_s[[0 \le d_s] \wedge [d_s \sim c] \wedge [PT'(s,f_2,d_s) \vee [PT'(s,f_1,d_s) \wedge P_s \wedge \bigvee_{i\in I(s)} [P_i \wedge PT(s_i,f_1 EU_{\sim(c-d_s)} f_2)]]]]$$

$$PT'(s,f_1 EU_{\sim c} f_2,d_s) \stackrel{\text{def}}{=} [[0 \le d_s] \wedge [d_s \sim c] \wedge [PT'(s,f_2,d_s) \vee [PT'(s,f_1,d_s) \wedge P_s \wedge \bigvee_{i\in I(s)} [P_i \wedge PT(s_i,f_1 EU_{\sim(c-d_s)} f_2)]]]]$$

図 6: アルゴリズム $PT(s,f)$

となる。逆にこの条件式を満たさない ρ に対しては上の条件は満たされない。したがって、この条件式は $(s,\rho) \models \langle a\rangle_{\sim c} f$ であるための必要十分条件となり、$PT(s,\langle a\rangle_{\sim c} f) = \exists d_s[0 \le d_s \wedge d_s \sim c \wedge P_s \wedge \bigvee_{i\in I}[P_i \wedge PT(s_i,f)]]$ となる。$PT(s,[a]_{\sim c} f)$ は、$[a]_{\sim c} f$ が $\neg\langle a\rangle_{\sim c}\neg f$ と同じ意味であることから直ちに導かれる。

$PT(s,f_1 EU_{\sim c} f_2)$ は次のようにして得られる。まず、$(s,\rho) \models f_1 EU_{\sim c} f_2$ を満たす付値を ρ としたとき、定義 3.2 より $(s,\rho) = (s_1,\rho_1) -t_1\rightarrow (s'_1,\rho'_1) -a_1\rightarrow \cdots -t_{k-1}\rightarrow (s'_{k-1},\rho'_{k-1}) -a_{k-1}\rightarrow (s_k,\rho_k)$ なるある遷移系列が存在して、$(s_k,\rho_k) \models f_2$ かつ $t_1 + \cdots + t_{k-1} \sim c$ かつ任意の i $(1 \le i \le k-1)$ に対して $(s_i,\rho_i) \models f$ が成り立つ。もし $t_1 \sim c$ かつ $(s,\rho) \models f_2$ ならば,$k=1$ とおけばよい。さもなければ、$(s,\rho) \models f_1$ かつ $(s,\rho) = (s_1,\rho_1) -t_1\rightarrow (s'_1,\rho'_1) -a_1\rightarrow (s_2,\rho_2)$ なるある系列が存在して $(s_2,\rho_2) \models f_1 EU_{\sim(c-t_1)} f_2$ を満たせばよい。$t_1 \sim c$ かつ $(s,\rho) \models f_2$ であるための ρ が満たすべき条件は

$$\rho \models \exists d_s[0 \le d_s \wedge d_s \sim c \wedge PT'(s,f_2,d_s)] \quad (1)$$

となる。ただし、$PT'(s,f_2,d_s)$ は $PT(s,f_2)$ の先頭の量化子 $\exists d_s$ を除去した条件式であり、一般には定義 4.2 に示す通りである。$(s,\rho) \models f_1$ である条件も同様に

$$\rho \models \exists d_s[0 \le d_s \wedge d_s \sim c \wedge PT'(s,f_1,d_s)] \quad (2)$$

となる。一方、$(s,\rho) = (s_1,\rho_1) -t_1\rightarrow (s'_1,\rho'_1) -a_1\rightarrow (s_2,\rho_2)$ なるある系列が存在して $(s_2,\rho_2) \models f_1 EU_{\sim(c-t_1)} f_2$ となるための ρ が満たすべき条件は次の通りである。まず、$(s,\rho) = (s_1,\rho_1) -t_1\rightarrow (s'_1,\rho'_1)$ より $s\text{-}e(d_s)[P_s]\rightarrow s'_1$ なる A-TSLTS の遷移が存在して $\rho \models \exists d_s[P_s]$ かつ $\rho'_1 = \rho[d_s = t_1]$ である。また、$(s'_1,\rho'_1) -a_1\rightarrow (s_2,\rho_2)$ より $s'_1\text{-}a_i[P_i]\rightarrow s_2$ なる A-TSLTS の遷移が存在して $\rho[d_s = t_1] \models P_i$ かつ $\rho_2 = \rho[d_s = t_1]$ となる。最後に、$(s_2,\rho_2) \models f_1 EU_{\sim(c-t_1)} f_2$ より $\rho[d_s = t_1] \models PT(s_2,f_1 EU_{\sim(c-d_s)} f_2)$ である。まとめると、

$$\rho \models \exists d_s[0 \le d_s \wedge d_s \sim c \wedge P_s \wedge P_i \wedge PT(s_2,f_1 EU_{\sim(c-d_s)} f_2)]$$

遷移 $s'_1\text{-}a_i[P_i]\rightarrow s_2$ の選び方は一般に複数の可能性がある。それらの論理和をとると

$$\rho \models \exists d_s[0 \le d_s \wedge d_s \sim c \wedge P_s \wedge \bigvee_{i\in I}[P_i \wedge PT(s_i,f_1 EU_{\sim(c-d_s)} f_2)]] \quad (3)$$

(ただし、$I \stackrel{\text{def}}{=} \{i | s'_1\text{-}a_i[P_i]\rightarrow s_i\}$) となる。以上より、$(s,\rho) \models f_1 EU_{\sim c} f_2$ となるために必要十分な ρ の条件は (1) または ((2) かつ (3))、すなわち、

$$\rho \models \exists d_s[0 \le d_s \wedge d_s \sim c \wedge [PT'(s,f_2,d_s) \vee [PT'(s,f_1,d_s) \wedge P_s \wedge \bigvee_{i\in I}[P_i \wedge PT(s_i,f_1 EU_{\sim(c-d_s)} f_2)]]]]$$

となる。

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 $PT(s,f)$ はパラメータ条件導出問題の解を与える。すなわち、

$$
\begin{aligned}
PT(s_0,f) &= \forall d[[0 \le d \wedge true] \Rightarrow [true \Rightarrow PT(s_1,f_1)]]\\
PT(s_1,f_1) &= \forall d_1[[0 \le d_1 \wedge d_1 < q_1 \wedge d_1 < p_1] \Rightarrow [d_1 < p_2 \Rightarrow PT(s_2,f_2)]]\\
PT(s_2,f_2) &= \exists d_2[0 \le d_2 \wedge d_2 \ge q_2 \wedge\\
&\quad [PT'(s_2,f_4,d_2) \vee [PT'(s_2,f_3,d_2) \wedge d_2 < p_3 \wedge d_2 < p_4 \wedge PT(s_3,f_2\{(q_2-d_2)/q_2\})]]]\\
PT'(s_2,f_3,d_2) &= [0 \le d_2 \wedge d_2 < p_3 \wedge d_2 < p_4 \wedge true]\\
PT'(s_2,f_4,d_2) &= [0 \le d_2 \wedge d_2 < p_3 \wedge p_4 \le d_2 \wedge true]\\
PT(s_3,f_2\{(q_2-d_2)/q_2\}) &= \exists d_3[0 \le d_3 \wedge d_3 \ge q_2 \wedge PT'(s_3,f_4,d_3)\\
&\quad \vee[PT'(s_3,f_3,d_3) \wedge d_3 < p_5 \wedge [[d_3 < p_6 \wedge PT(s_4,f_2\{(q_2-d_2-d_3)/q_2\})]]\\
&\quad \vee[d_3 \ge p_6 \wedge PT(s_5,f_2\{(q_2-d_2-d_3)/q_2\})]]]]]\\
PT'(s_3,f_3,d_3) &= [0 \le d_3 \wedge d_3 < p_5 \wedge false] \quad \equiv false\\
PT'(s_3,f_4,d_3) &= [0 \le d_3 \wedge d_3 < p_5 \wedge false] \quad \equiv false\\
PT(s_3,f_2\{(q_2-d_2)/q_2\}) &\equiv false\\
PT(s_2,f_2) &\equiv \exists d_2[0 \le d_2 \wedge q_2 \le d_2 \wedge [[0 \le d_2 \wedge d_2 < p_3 \wedge p_4 \le d_2 \wedge true]\\
&\quad \vee[[0 \le d_2 \wedge d_2 < p_3 \wedge d_2 < p_4 \wedge true] \wedge d_2 < p_3 \wedge d_2 < p_4 \wedge false]]]\\
&\equiv [0 < p_3 \wedge p_4 < p_3 \wedge q_2 < p_3]\\
&\equiv [0 < p_3 \wedge p_4 < p_3 \wedge q_2 < p_3]\\
PT(s_1,f_1) &\equiv \forall d_1[[0 \le d_1 \wedge d_1 < q_1 \wedge d_1 < p_1] \Rightarrow [d_1 < p_2 \Rightarrow [0 < p_3 \wedge p_4 < p_3 \wedge q_2 < p_3]]]\\
&\equiv [0 < q_1 \wedge 0 < p_1 \wedge 0 < p_2] \Rightarrow [0 < p_3 \wedge p_4 < p_3 \wedge q_2 < p_3]\\
PT(s_0,f) &\equiv [0 < q_1 \wedge 0 < p_1 \wedge 0 < p_2] \Rightarrow [0 < p_3 \wedge p_4 < p_3 \wedge q_2 < p_3]
\end{aligned}
$$

図 7: パラメータ条件の導出例

$$\forall \rho[\rho \models PT(s,f) \Leftrightarrow (s,\rho) \models f] \qquad \square$$

例 4.1 例 2.3 の動作仕様の初期状態 s_0 が例 3.1 の要求仕様のうち

(1) $[press][release]_{<q_1}$
$((\langle press\rangle true) EU_{\ge q_2} (\langle shortsingleclick\rangle true))$

を満たすためのパラメータ条件を定義 4.2 のアルゴリズムで導出したときの導出過程を図 7 に示す。図 7 では、記述の便宜のため上の TCTL 式を f として、f の各部分式をそれぞれ $f = [press]f_1$, $f_1 = [release]_{<q_1}f_2$, $f_2 = f_3 EU_{\ge q_2} f_4$, $f_3 = (\langle press\rangle true)$, $f_4 = (\langle shortsingleclick\rangle true))$ とおき、$f\{e/x\}$ で式 f のなかの変数 x を式 e で置き換えた式を表している。導出結果より、要求仕様のパラメータ q_1,q_2 に対して、$[0 < q_1 \wedge 0 < p_1 \wedge 0 < p_2] \Rightarrow [0 < p_3 \wedge p_4 < p_3 \wedge q_2 < p_3]$ を満たすように動作仕様のパラメータ値 p_1〜p_7 を決めれば、要求仕様の (1) 式を満足できることが分かる。 $\square$

5 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。一般に A-TSLTS がループを含む場合、関数 $PT(s, f_1 EU_{\sim c} f_2)$ 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。ただし、導出される式は一般に量化子を含む式になるため、手で人間が理解しやすい式に簡約化する必要がある。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、および、式の簡約化を自動化することは今後の課題である。

謝辞

本研究は (財) 電気通信普及財団の助成により行われました。

参考文献

[1] 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).

[2] Alur, R., Courcoubetis, C. and Dill, D.: Model-Checking in Dense Real-Time, *Information and Computation*, Vol. 104, pp. 2–34 (1993).

[3] Wang, F.: Parametric Timing Analysis for Real-Time Systems, *Information and Computation*, Vol. 130, No. 2, pp. 131–150 (1996).

[4] Alur, R. and Dill, D.: Automata For Modelling Real-Time Systems, in *Proc. of ICALP'90*, Vol. 443 of *Lecture Notes in Computer Science*, pp. 322–335, Springer-Verlag (1990).