HAL open science

Completeness and the expressive power of next time temporal logical system by semantic tableau method

O. Katai

▶ To cite this version:

O. Katai. Completeness and the expressive power of next time temporal logical system by semantic tableau method. RR-0109, INRIA. 1981. inria-00076451

HAL Id: inria-00076451
https://inria.hal.science/inria-00076451

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CENTRE DE ROCQUENCOURT

Rapports de Recherche

N° 109

COMPLETENESS AND THE EXPRESSIVE POWER OF NEXTTIME TEMPORAL LOGICAL SYSTEM BY SEMANTIC TABLEAU METHOD

Osamu KATAI

Décembre 1981

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél. (3) 954 90 20

COMPLETENESS AND THE EXPRESSIVE POWER OF NEXTTIME TEMPORAL LOGICAL SYSTEM BY SEMANTIC TABLEAU METHOD

Osamu KATAI

RESUME Dans ce papier on étudie la complétude et le pouvoir expressif du système à logique temporelle de Pnueli. On précise la classe des ω-langages représentable par un tel système.

ABSTRACT The completeness and the expressive power of Pnueli's nexttime temporal logical system is investigated and the class of ω-languages representable by this system is clarified.

Kyoto University, JAPON.

1. Introduction

Temporal logical systems[1],[2],[3] provide a quite natural and simple way for the verification of programs, particularly for that of concurrent programs, in which the notion of time invariance and causality play crucial roles.

In this paper, we investigate the completeness and the expressive power of Pnueli's nexttime temporal logical system DX[2] (or the propositional part of the nexttime system in [3]), which is an augmented version of linear time temporal logical system K_1[4] by incorporating it with a new tense operator X representing the next time instant. The axioms and rules of DX are as follows together with those of Propositional Calculus.

A1: $\vdash X(A \supset B) \supset (XA \supset XB)$

A2: $\vdash XA \supset \sim X \sim A$

A3: $\vdash \sim X \sim A \supset XA$

A4: $\vdash G(A \supset B) \supset (GA \supset GB)$

A5: $\vdash GA \supset (A \wedge XGA)$

R1: if $\vdash A$, then $\vdash XA$.

R2: if $\vdash A \supset XA$, then $\vdash A \supset GA$,

where A and B are arbitrary propositions(<u>temporal formulae</u>), and $A \supset B$ stands for $\sim(A \wedge \sim B)$(in the sequel, we use only $\wedge$ and $\sim$ as primitive symbols for logical connectives). GA represents that A holds <u>forever</u>(including the present time) and XA represents that A holds <u>at the next time instant</u>. $\sim G \sim A$, often denoted as FA, can be interpreted as A holds at <u>sometime</u> in the future.

In order to discuss the <u>infinite sequences</u>(<u>ω-seq.'s</u>) of events represented by these temporal formulae, we refer to the theory of <u>ω-automata</u>[5],[6]. These (finite) automata <u>$M(s_0, S, \mathcal{T})$</u> accept an input ω-seq. iff the set of limiting states(the states entered infinitely often when drived by the input seq. from initial state s_0) coincides with one of the designated family $\mathcal{T}$ of subsets of S called <u>anchored sets</u>. For the following discussion, it is convenient to use the notion, <u>macrosource</u>, which is a nondeterministic ω-automata with possibly partially defined transitions and with not necessarily singular initial states[5]. Moreover, we label input alphabets not to <u>arrows</u>(transitions between states) but instead to states themselves. Hence the modified ones can be regarded not as <u>acceptors</u> but instead as <u>generators</u> of ω-seq.'s. For example, the following macrosource $M(S_0, S, \mathcal{T})$ generates

$$\bigcup_{i:\text{ odd no.}} \underbrace{pp\cdots p}_{i} \sim p \sim p \cdots \sim p \cdots.$$

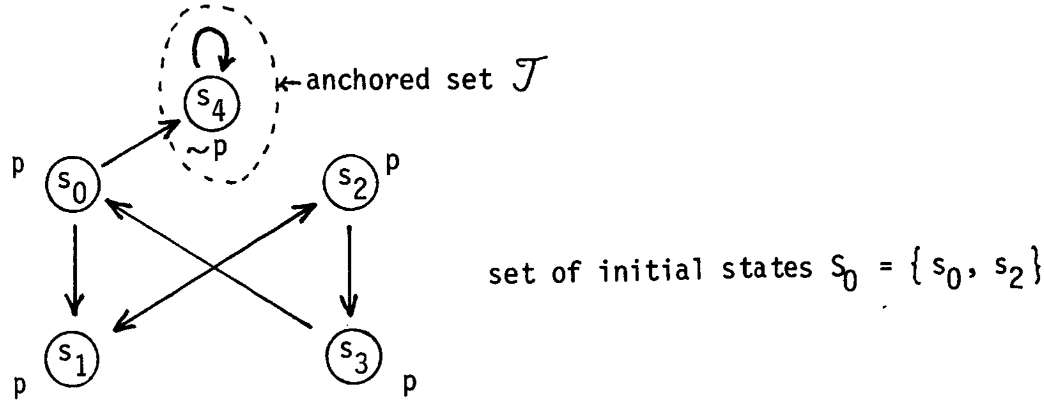


Fig. 1. Macrosource generating $\bigcup_{i:odd} \underbrace{pp \cdots p}_{i} \sim p \sim p \cdots \sim p \cdots.$

It can be readily seen that these modifications have no effect on the expressive power of ω-automata, and any $\underline{\omega\text{-language}}$(i.e. set of ω-seq.'s) which can be generated by a macrosource $M(S_0, S, \mathcal{T})$ is called $\underline{\omega\text{-regular}}$ and is denoted by $\underline{L(S_0, S, \mathcal{T})}$.

2. Model of DX and ω-Language

Let $\underline{sub(H)}$ be the set of subformulae in a proposition H. Then a $\underline{\text{model}}$ of H is an ω-seq. $\mathcal{F}_0, \mathcal{F}_1, \cdots, \mathcal{F}_t, \cdots$ of subsets of sub(H) s.t. for every t,

if $\sim A \in \mathcal{F}_t$, $A \notin \mathcal{F}_t$; if $A \notin \mathcal{F}_t$ and $\sim A \in sub(H)$, $\sim A \in \mathcal{F}_t$,

if $A \wedge B \in \mathcal{F}_t$, $A, B \in \mathcal{F}_t$; if $A, B \in \mathcal{F}_t$ and $A \wedge B \in sub(H)$, $A \wedge B \in \mathcal{F}_t$,

if $XA \in \mathcal{F}_t$, $A \in \mathcal{F}_{t+1}$; if $A \in \mathcal{F}_{t+1}$ and $XA \in sub(H)$, $XA \in \mathcal{F}_t$,

if $GA \in \mathcal{F}_t$, $A \in \mathcal{F}_{t'}$ for $\forall t' \geq t$; if $A \in \mathcal{F}_{t'}$ for $\forall t' \geq t$ and $GA \in sub(H)$, $GA \in \mathcal{F}_t$,

and also

$H \in \mathcal{F}_0$.

It can be readily seen that each model of H uniquely corresponds to an ω-seq. $\xi = (\xi^0, \xi^1, \cdots, \xi^t, \cdots)$ of $\underline{\text{state descriptions}}$(conjunction of every propositional variable or its negation in H) by giving ξ^t as $\bigwedge_{p \in P(H) \cap \mathcal{F}_t} p \wedge \bigwedge_{p \in P(H) - \mathcal{F}_t} \sim p$, where P(H) is the set of propositional variables in H. We denote by $\underline{L(H)}$ the set of such ω-seq's. By regarding each state description(in P(H)) as an alphabet, L(H) can be regarded as an ω-language.

3. Construction of Semantic Tableau(Transition Diagram) for Temporal Formulae

$\underline{\text{Semantic tableau methods}}$ provide a systematic way to search for every possible model of an arbitrarily given proposition H[7],[8]. Apart from the usual method for modal logical systems introduced by Kripke[8], we construct a new semantic tableau which is a kind of transition diagram(trans. diag.), i.e., directed graph consisted of nodes called tableaux(tab.'s) and of directed lines(arrows) between them. Each tab. s_i represents a state(time instant) and has right and left columns(col.'s) each of which consists of certain propositions. The left col. $L(s_i)$ represents the propositions holding at that state and the right col. $R(s_i)$ represents those not holding at that time(we denote s_i as $\{L(s_i); R(s_i)\}$). Each arrow s_iXs_j represents that state s_j comes next(at the next time instant) to state s_i. The rules of its construction are as in the sequel, where A and B are arbitrary propositions and s_i stands for an arbitrary tab. at any stage of the construction.

(Init.) Put a tab., say s_0, with only H in the left col.(the right col. being vacant) and call it "the main tab.", i.e., $s_0 = \{ H; \}$.

(N) If $\sim A$ appears in the left(right) col; of s_i, put A in the right(left) col. of s_i.

($\wedge$l) If $A \wedge B$ appears in the left col. of s_i, put A and B in that col.

($\wedge$r) If $A \wedge B$ appears in the right col. of s_i, make two copies $s_{i,1}$ and $s_{i,2}$ of s_i, draw arrows $s_jXs_{i,1}$ and $s_jXs_{i,2}$(or $s_{i,1}Xs_j$ and $s_{i,2}Xs_j$) for every s_j s.t. s_jXs_i(or s_iXs_j) and erase s_i from the trans. diag. Moreover, put A in the right col. of $s_{i,1}$ and put B in the right col. of $s_{i,2}$.

(X) If XA appears in the left(right) col. of s_i, put A in the left(right) col. of every $s_j \in X(s_i)$, where $X(s_i) \overset{\text{df.}}{=} \{ s_j | s_iXs_j \}$. If $X(s_i) = \phi$, make a new tab. $s_j = \{ A; \}$ ($\{ ; A \}$) and draw arrow s_iXs_j.

(Gl) If GA appears in the left col. of s_i, put A in the left col. of s_i and put GA in the left col. of $\forall s_j \in X(s_i)$. If $X(s_i) = \phi$, make a new tab. $s_j = \{ GA; \}$ and draw arrow s_iXs_j.

(Gr) If GA appears in the right col. of s_i, make two copies $s_{i,1}$ and $s_{i,2}$ of s_i(and erase s_i), put A in the right col. of $s_{i,1}$ and also in the left col. of $s_{i,2}$. Make a copy $s_{j'}$ of s_j for every $s_j \in X(s_i)$ and put GA in its right col. Moreover, draw arrows $s_{i,1}Xs_j$, $s_{i,2}Xs_{j'}$, and $s_{j'}Xs_k$ for $\forall s_j \in X(s_i)$ and $\forall s_k \in X(s_j)$. If $X(s_i) = \phi$, make a new tab. $s_j = \{ ; GA \}$ and draw arrow $s_{i,2}Xs_j$.

Rules (Gl) and (Gr) correspond to axiom A5 and $\vdash FA \supset (A \vee (\sim A \wedge XFA))$, respectively.

(Mer.) If s_i and s_j are identical, i.e. $L(s_i) = L(s_j)$ and $R(s_i) = R(s_j)$, and all the possible operations above for $\forall s_k$ s.t. $s_kX^*s_i$ or $s_kX^*s_j$(X^* represents the transitive closure of directed relation(arrow) X) have been done, merge

s_i and s_j(i.e., erase s_j and draw arrows $s_k X s_i$(or $s_i X s_k$) for $\forall s_k$ s.t. $s_k X s_j$ (or $s_j X s_k$)).

(Proc.) The operations above have no priority to each other.

(Ter.) If a closed tab.(see below) appears, stop the operation to the tab. If all the possible operations to every open tab. have been done, the construction of the diagram terminates.

Def. 1: If s_i has a formula common in both sides of col.'s, i.e., $L(s_i) \cap R(s_i) \neq \phi$, s_i is called closed; otherwise it is called open[7]. If s_i is consisted of only non-temporal formulae(i.e. formulae containning neither X nor G), it is called free.

Theorem 1: The construction of the trans. diag. for any H terminates in a bounded number(but depending on H) of operations, and hence the final diag. is finite.

Proof: It is evident that the formulae in the tab.'s at any stage of the construction is contained in sub(H). Hence, by operation (Mer.), the number of possible different trans. diag. is bounded. □

For technical convenience, we supplement new tab.'s $s_{f,1} = \{p; \ \}$ and $s_{f,2} = \{ \ ; p\}$ with $s_{f,i} X s_{f,j}$($i = 1,2$; $j = 1,2$) to the final transition diagram S and draw arrows $s_i X s_{f,1}$ and $s_i X s_{f,2}$ for every free and open tab. s_i in S, where p is an arbitrary propositional variable in H.

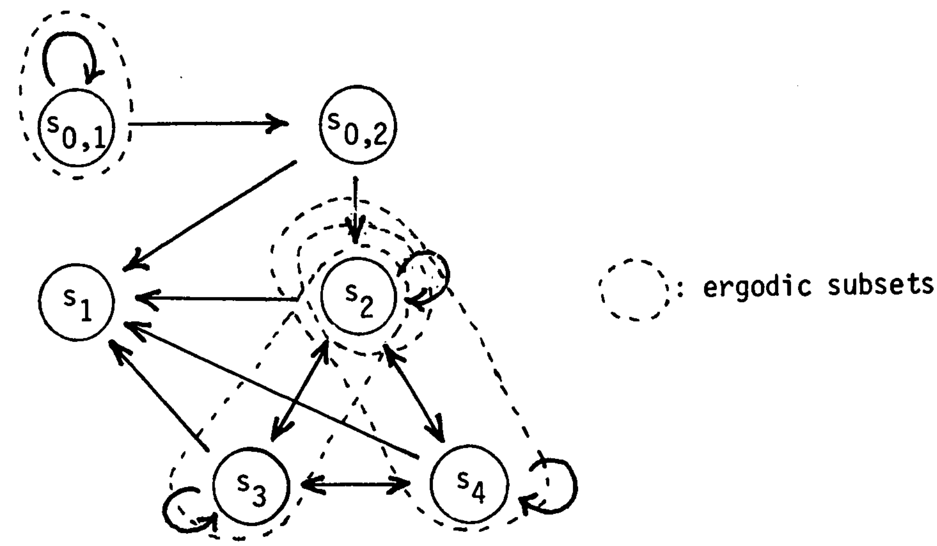
There exist two kinds of inconsistencies in our trans. diag. which obstruct the interpretation(model construction) for H. The "closedness" in Def. 1 represents a kind of "static" inconsistency at a time instant; the "transientness" introduced below stands for a kind of "dynamic" inconsistency.

Def. 2: An open tab. s_i in S is called X-transient with respect to a subset S' of S iff $S' \cap X(s_i) = \phi$. An open and non X-transient tab. s_i is called G-transient w.r.t. S' iff there exists a proposition A s.t. GA is contained in the right col. of s_i and A is not contained in the right col. of any s_j s.t. $i = j$ or $d_i X^* s_j$.

These definitions say that if one enter a state represeted by s_i, one must go out of S' in the future, i.e., one cannot remain in S' forever.

Def. 3: A subset S' of S is called ergodic iff every tab. in S' is open and also is neither X-transient nor G-transient w.r.t. S'.

For example, the trans. diag. of proposition $G(\sim(p \wedge XG\sim p))(\equiv G(p \supset XFp))$ is given as in Fig. 2, where $s_{0,1}$ and $s_{0,2}$ are the main tab.'s(devided by rule $(\wedge r)$) and the ergodic subsets are as shown in the figure, for s_1 is closed and $\sim p$ is not contained in the right col.'s of s_3 and s_4 while $G\sim p$ is contained in those col.'s.



$s_{0,1}$:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	p

$s_{0,2}$:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	$XG\sim p$

s_1:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	$G\sim p$
p	$\sim p$
	p

s_2:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	$G\sim p$
p	$\sim p$
	$XG\sim p$

s_3:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	p
$\sim p$	$G\sim p$

s_4:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	$G\sim p$
$\sim p$	p
	$XG\sim p$

Fig. 2. Semantic tableau(transition diagram) for proposition $G(\sim(p \wedge XG\sim p))$.

4. Completeness of DX and the Class of ω-Languages Representable by Temporal Formulae

In this section, we investigate the language class of L(H) introduced in the final part of section 2.

Theorem 2: L(H) is ω-regular for any proposition H.

Proof: L(H) can be generated by the macrosource $M(S_0, S, \mathcal{T})$ given as follows, where S is the trans. diag. of H. To each tab. s_i in S, we label state descriptions $\xi_{i_1}, \xi_{i_2}, \ldots, \xi_{i_{n_i}}$ s.t. $\xi_{i_1} \vee \xi_{i_2} \vee \ldots \vee \xi_{i_{n_i}} \equiv A'(s_i)$, where $A'(s_i)$ is defined as $\bigwedge_{P \in L'(s_i)} P \wedge \bigwedge_{P \in R'(s_i)} \sim P$, and $L'(s_i)$ and $R'(s_i)$ are the sets of non-temporal formulae in $L(s_i)$ and $R(s_i)$, respectively. The set S_0 of initial states is given as $\{s_0\}$ if the main tab. s_0 is not devided by rules ($\wedge$r) or (Gr) or as the set $\{s_{0,1}, s_{0,2}, \cdots, s_{0,n_0}\}$ of main tab.'s if it is devided. The family of anchored sets is given as $\mathcal{T} = \{$ ergodic subsets of $S\}$. □

Moreover, from our semantic tableau method, we can show that

Theorem 3(completeness of DX): If the trans. diag. of H has no ergodic subsets, then $\sim H$ is provable in temporal logical system DX.

Proof: It is done rather differently from Kripke's method[8] for the completeness proof of modal logical systems. However, the following formulae play similar roles as those in his method.

Def. 4: We call the following formulae $A(s_i)$ and $C(s_i)$ the associated formula and the characteristic formula of tab. s_i, respectively.

$$A(s_i) \overset{df.}{=} \bigwedge_{P \in L(s_i)} P \wedge \bigwedge_{P \in R(s_i)} \sim P$$

$$C(s_i) \overset{df.}{=} A(s_i) \supset X\Big(\bigvee_{s_j \in X(s_i)} A(s_j)\Big) \quad \Big(\equiv \ \sim\big(A(s_i) \wedge X\big(\bigwedge_{s_j \in X(s_i)} \sim A(s_j)\big)\big)\Big)$$

The main gist of the proof is that $C(s_i)$ is provable in DX for any s_i at any stage of the construction. □

For the discussion in the sequel, we also need the next formula.

Def. 5: For the final trans. diag. $S = \{s_0, s_1, \cdots, s_n\}$, we call the follwing formula E_i the extended characteristic formula of tab. s_i(in S), which contains s_0, s_1, $\cdots$, and s_n as supplemented propositional variables,

$$E_i(S; s_0, s_1, \cdots, s_n) \overset{df.}{=} s_i \wedge \bigwedge_{s_i \in S} \Big\{ G\Big(s_i \supset X\big(\bigvee_{s_j \in X(s_i)} s_j\big)\Big) \wedge G(s_i \supset A(s_i)) \Big\}.$$

We then consider the problem: "what kind of ω-regular language is representable by temporal formulae".

Def. 6: For arbitrary family of ω-languages L_0, L_1, $\cdots$, and L_n, we call them temporally distinguishable from each other iff there exist propositions A_0, A_1, $\cdots$, and A_n satisfying

$$L(A_i) \supset L_i \quad \text{for } i = 0, 1, \cdots, n,$$

$$\vdash \sim(A_i \wedge A_j) \quad \text{for } \forall i \neq \forall j.$$

The next theorem provides a sufficient condition for temporal representability.

Theorem 4: For an arbitrary ω-regular language $L = L(S_0, S, \mathcal{T})(S = \{ s_0, s_1, \cdots, s_n \})$, if $L(s_i, S, \mathcal{T})(i = 0, 1, \cdots, n)$ are temporally distinguishable from each other, then L is temporally representable, i.e., there exists a proposition B s.t. $L(B) = L$.

Proof: Let A_i's be as in Def. 6. Then B can be given as

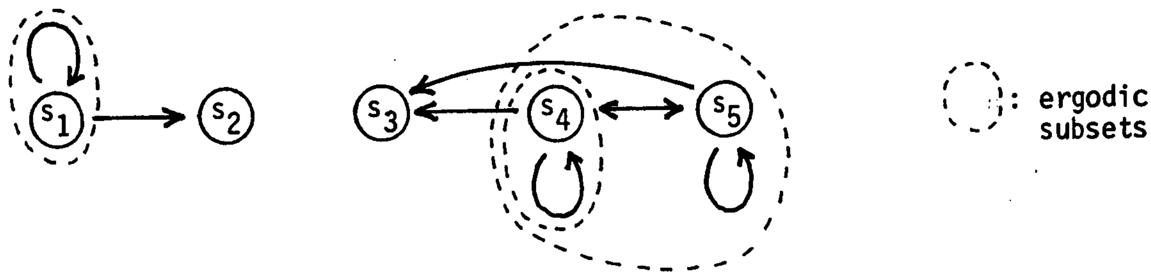
$$B \equiv \bigvee_{s_i \in S_0} E_i(S; A_0, A_1, \cdots, A_n). \Box$$

By introducing an extended version of our semantic tableau, we can show that the converse of the above theorem also holds.

Theorem 5: For any formula H, there exist a macrosource $M(S_0, S, \mathcal{T})$ such that $L(S_0, S, \mathcal{T}) = L(H)$ and $L(s_i, S, \mathcal{T})(i = 0, 1, \cdots, n)$ are temporally distinguishable from each other(we call such macrosource temporally distinguishable macrosource).

Proof: We introduce the following semantic tableau method: Make every possible division of sub(H) into two subsets L(H) and R(H). Then we make tab.'s { L(H); R(H) } (thus we have $2^{\#\text{sub}(H)}$ tab.'s), and put them as the initial transition diagram(without any arrows). Apply the operations (N)$\sim$(Gr) and (Mer.) to draw arrows among them and also to delete inconsistent or superfluous tab.'s(if a new tab. appears in the diagram, it is certainly a closed tab., and hence it can be removed from the diagram). From the final trans. diag. S, we make the macrosource $M(S_0, S, \mathcal{T})$ as $S_0 = \{ s_i \mid H \in L(s_i)\}$ and $\mathcal{T} = \{$ ergodic subsets of S $\}$. M is a temporally distinguishable macrosource, for the associated formulae $A(s_i)$'s satisfy the condition on A_i's in Def. 6. Also, it can be readily seen that it generates language L(H). $\Box$

For example, the extended version of Fig. 2 is given as Fig. 3, where all the tab.'s are the main tab.'s, and the three ergodic subsets show that $L(G{\sim}(p \wedge XG{\sim}p))$ is equal to the disjoint union of $L(G{\sim}p)$, $L(Gp)$ and $L(GFp \wedge GF{\sim}p)$.



s_1:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	p
$\sim p$	
$XG\sim p$	
$G\sim p$	

s_2:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	p
$\sim p$	$XG\sim p$
$G\sim p$	

s_3:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	p
$\sim p$	$G\sim p$
$XG\sim p$	

s_4:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	$XG\sim p$
p	$\sim p$
	$G\sim p$

s_5:

$G(\sim(p \wedge XG\sim p))$	$p \wedge XG\sim p$
$\sim(p \wedge XG\sim p)$	p
$\sim p$	$XG\sim p$
	$G\sim p$

Fig. 3. Extended semantic tableau(trans. diag.) for $G(\sim(p \wedge XG\sim p))$.

From the above two theorems, we finally obtain

<u>Theorem 6</u>: The class of ω-languages representable by temporal formulae coincides with that of ω-regular languages which can be generated by temporally distiguishable macrosources.

The above class has not so wide variety. For example, we have

<u>Theorem 7</u>: The language in Fig. 1 is not temporally representable.

Proof: We make an equivalent ω-automaton $M(\bar{s}_0, \bar{S}, \bar{\mathcal{T}})$(macrosource with deterministic and fully defined transitions and singular initial state) to the macrosource in Fig. 1.

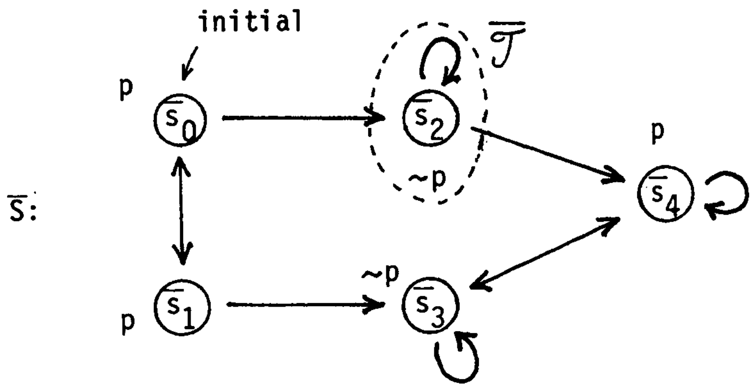


Fig. 4. An equivalent ω-automaton to the macrosource in Fig. 1.

Then we have the following lemmas.

Lemma 1: If $L_0 = L(\bar{s}_0, \bar{S}, \bar{\mathcal{J}})$ is temporally representable, so is $L_i = L(\bar{s}_i, \bar{S}, \bar{\mathcal{J}})$ for any $\bar{s}_i$ accessible from $\bar{s}_0$.

Lemma 2: $L_0 \cap L_1 = \phi$.

Lemma 3: If L_0 and L_1 are temporally distinguishable from each other, we have

$$\vdash \sim(E_0(S; A_0, A_1, \cdots, A_4) \wedge E_1(S; A_0, A_1, \cdots, A_4))$$

for any formulae $A_0 \sim A_4$.

However, the following instance contradicts the above conclusion: $A_0 \equiv A_4 \equiv p$, $A_1 \equiv p \wedge Tp$ and $A_2 \equiv A_3 \equiv \sim p$. Therefore, L_0 is not temporally representable. □

5. Concluding Remarks

We have clarified(as summarized in Theorem 6) the class of ω-languages representable by Pnueli's temporal logical system DX. It has turned out that the class has no close relationships with the nice classification of ω-regular languages in terms of topology in the space of ω-seq.'s discussed by Landweber et al.[6]. The class being merely a small subclass of ω-regular languages, we certainly need some kind of reinforcement to DX for the treatment of complex systems such as programs. The method used so far is to incorporate DX certain special propositional variables representing the positions in the flow charts of programs which are ready to be executed[2],[3]. This supplementation seems, from theoretical point of view, to be unfinished, and we need a more general framework of temporal logical systems for the treatment of programs.

REFERENCES

[1] Kröger, F.: A uniform logical basis for the description, specification and verification of programs, in E. J. Neuhold(ed.): Formal Description of Programming Concepts, pp.441 - 459, North-Holland(1978).

[2] Pnueli, A.: The temporal semantics of concurrent programs, in G. Kahn(ed.): Semantics of Cocurrent Computation, Lecture Notes in Computer Science, Vol. 70, pp.1 - 20, Springer(1979).

[3] Manna, Z. and Pnueli, A.: The temporal logic of programs, in H. A. Maurer (ed.): Automata, Languages and Programming, Lecture Notes in Computer Science, Vol. 71, pp.385 - 409, Springer(1979).

[4] Rescher, N. and Urquhart, A.: Temporal Logic, pp.1 - 97, Springer(1971).

[5] Trakhtenbrot, B. A. and Barzdin, Ya, M.: Finite Automata(Engl. Trans.), pp.1 - 66, North-Holland(1973).

[6] Wagner, K.: On ω-regular sets, Information and Control, Vol. 43, pp.123 - 177(1979).

[7] Beth, E. W.: Semantic entailment and formal derivability, Mededelingen der Kon. Nederlandse Akad. Wetensch. Afdeling. Letter., Vol. 18, No. 13, pp.309 - 342(1955).

[8] Kripke, S. A.: Semantical analysis of modal logic I: Normal modal propositional calculi, Zeitschrift fur Math. Logik und Grund. der Mathematik, Vol. 9, pp.67 - 96(1963).