

命题线性时序逻辑的对偶模型问题的复杂性*

吴志林⁺, 张文辉

(中国科学院 软件研究所 计算机科学重点实验室, 北京 100080)

The Complexity of Dual Models Problem of Propositional Linear Temporal Logics

WU Zhi-Lin⁺, ZHANG Wen-Hui

(State Key Laboratory of Computer Science, Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

+ Corresponding author: Phn: +86-10-62661616, Fax: +86-10-62661627, E-mail: wuzl@ios.ac.cn

Wu ZL, Zhang WH. The complexity of dual models problem of propositional linear temporal logics. Journal of Software, 2007,18(7):1573-1581. <http://www.jos.org.cn/1000-9825/18/1573.htm>

Abstract: In this paper, the concept of dual models of a propositional linear temporal logic formula is defined: A formula f has dual models if it has two models (namely two ω -sequences of states) such that the assignments to atomic propositions at each position of them are dual. Then for various propositional linear temporal logics, the complexity of the problem deciding whether a formula f has dual models (denoted by DM) and the problem of determination of dual models in a Kripke-structure for a formula f (denoted by KDM) are investigated. It is shown that DM and KDM are NP-complete for the logic with F ("Future") operator, and they are PSPACE-complete for the logic with F, X ("Next") operators, the logic with U ("Until") operator, the logic with U, S, X operators, and the logic with regular operators given by Wolper (known as extended temporal logic, ETL).

Key words: propositional linear temporal logic; dual model; computational complexity

摘要: 定义了一个命题线性时序逻辑的对偶模型的概念. 一个公式 f 的对偶模型是指 f 的满足以下条件的两个模型(即状态的 ω 序列): 在每个位置上这两个模型对原子命题的赋值都是对偶的. 然后, 对于确定一个公式 f 是否有对偶模型的判定问题(记为 DM)和在一个 Kripke-结构中确定是否存在从两个给定状态出发的对偶模型满足给定公式 f 的判定问题(记为 KDM)的复杂性进行了研究. 证明了以下结果: 对于只含有 F ("Future") 算子的命题线性时序逻辑, DM 和 KDM 都是 NP 完全的; 而对于以下命题线性时序逻辑, DM 和 KDM 都是 PSPACE 完全的: 含有 F, X ("Next") 算子的逻辑、含有 U ("Until") 算子的逻辑、含有 U, S, X 算子的逻辑以及由 Wolper 给出的含有正规语言算子的逻辑(一般称为扩展时序逻辑, 简称 ETL).

关键词: 命题线性时序逻辑; 对偶模型; 计算复杂性

中图法分类号: TP301 文献标识码: A

* Supported by the National Natural Science Foundation of China under Grant Nos.60223005, 60573012 (国家自然科学基金); the National Basic Research Program of China under Grant No.2002cb312200 (国家重点基础研究发展计划(973))

Received 2005-07-08; Accepted 2006-02-23

1 Introduction

Linear temporal logic was introduced in Ref.[1] as an appropriate formal system for reasoning about parallel programs and reactive systems. Linear temporal logic can be used to describe temporal properties of systems conveniently and briefly, such as the properties of deadlock free, liveness etc.

The complexity of satisfiability (SAT) and model checking (MC) problems of propositional linear temporal logics has been investigated extensively in Ref.[2–6] motivated by their applications to the synthesis of concurrent systems from specifications^[7,8] and the verification of concurrent and reactive systems^[9,10]. While all these investigations are from a practical view, we, from a theoretical view, are wondering whether there are other natural decision problems for propositional linear temporal logics. With the theoretical view in mind, we examine various variants of SAT problem of Boolean logic (such as NAESAT, MAXSAT, MAJSAT, UNIQUESAT etc^[11]) to see whether they can be generalized into propositional linear temporal logics. Finally we find out that among them, NAESAT can be generalized naturally and easily into propositional linear temporal logics, namely dual models problem (DM) defined in this paper.

NAESAT is to decide whether a boolean formula in 3-CNF is not-all-equal satisfiable: A boolean formula f in 3-CNF (conjunctive normal form such that each clause has exactly three literals) is not-all-equal satisfiable (NAESAT) if there is an assignment to atomic propositions such that each clause has at least one true literal and at least one false literal. The complexity of NAESAT is the same as SAT for boolean logic, namely NP-complete.

It is easily shown that a 3-CNF boolean formula f is not-all-equal satisfiable iff f has two models whose assignments to atomic propositions are dual. Then naturally, we can generalize this concept of duality of assignments to atomic propositions to propositional linear temporal logics and define so called dual models problem.

Dual models of a propositional linear temporal logic formula f are two models (namely two ω -sequences of states) of f such that the assignments to atomic propositions at each position of them are dual. Dual models problem (DM), as a variant of SAT for propositional linear temporal logic, is to decide whether a given formula f has dual models. Moreover, in view of model checking problem, we consider a similar problem, namely the problem of determination of dual models in a Kripke structure (KDM) defined as follows: KDM problem is to decide for a given formula f , a Kripke structure K and two states δ_1, δ_2 in K , whether there are two dual paths p, q starting from δ_1, δ_2 respectively such that they both satisfy f .

In Ref.[2], the complexity of SAT and MC was investigated for various propositional linear temporal logics. It was shown that SAT and MC are NP-complete for the logic with F (“Future”) operator, and are PSPACE-complete for the logic with F, X (“Next”) operators, the logic with U (“Until”) operator, the logic with U, S, X operators, and the logic with regular operators given by Wolper (known as Extended Temporal Logic, ETL). On the basis of those results for SAT and MC in Ref.[2], we investigate the complexity of DM and KDM in this paper and show that the complexity of DM and KDM is the same as that of SAT and MC for various propositional linear temporal logics. Moreover, our techniques used in this paper are general enough to determine the complexity of DM and KDM problems for all propositional linear temporal logics that admit complete Boolean operators. Consequently once we have determined the complexity of SAT and MC problems for a propositional linear temporal logic, we know the complexity of DM and KDM problems as well.

This paper is organized as follows. Section 2 defines the syntax and semantics of propositional linear temporal logic, and the related decision problems. In Section 3, we study some properties of the dual models. In Section 4, we investigate the complexity of DM and KDM problems. Finally in Section 5, we give some conclusions and remarks.

2 Notation and Definitions

In this paper, we follow the notation and definitions of Ref.[2].

2.1 Syntax and semantics of propositional linear temporal logics

A regular right linear grammar is a regular grammar in which all the production rules are of the form $N \rightarrow aM$, $N \rightarrow a$, where N, M are nonterminals in the grammar and a is a string of terminal symbols.

Given a set of atomic propositions P , and a regular right linear grammar G with terminal symbols a_1, \dots, a_n and nonterminal symbols N_1, \dots, N_m , the syntax of extended temporal logic (denoted by $ETL(G)$) is defined as follows:

$$f_i := P | \neg f_1 | f_1 \vee f_2 | \mathbf{X}f_1 | f_1 \mathbf{U}f_2 | f_1 \mathbf{S}f_2 | N_j(f_1, \dots, f_n), \text{ where } P \in P, 1 \leq j \leq m.$$

In addition, the following abbreviations are defined:

$$f_1 \wedge f_2 \equiv \neg(\neg f_1 \vee \neg f_2), f_1 \rightarrow f_2 \equiv \neg f_1 \vee f_2, \mathbf{F}f \equiv \text{True} \mathbf{U}f, \mathbf{G}f \equiv \neg \mathbf{F} \neg f.$$

Let $O_1, \dots, O_k \in \{\mathbf{X}, \mathbf{F}, \mathbf{G}, \mathbf{U}, \mathbf{S}\}$, then $L(O_1, \dots, O_k)$ denotes the sublogic of $ETL(G)$ restricted to these operators, for example $L(\mathbf{F})$, $L(\mathbf{F}, \mathbf{X})$, $L(\mathbf{U})$, $L(\mathbf{U}, \mathbf{X})$, $L(\mathbf{U}, \mathbf{S}, \mathbf{X})$ and so on. We denote the logic containing \mathbf{F}, \mathbf{X} operators but with “ \neg ” operators applied only to atomic propositions by $\tilde{L}(\mathbf{F}, \mathbf{X})$.

A structure $S = (s, \xi)$, where $s = (s_0, s_1, \dots)$, is an ω -sequence of states in which all the states are distinct and $\xi: \{s_0, s_1, \dots\} \rightarrow 2^P$. Intuitively, ξ specifies which atomic propositions are true in each state. An interpretation is a pair (S, δ) where S is a structure as previously defined, and δ is a state in the sequence s .

Semantics of $ETL(G)$ are defined as follows:

Let $S = (s, \xi)$ be a structure and s_i be a state in s , then

$$(S, s_i) \models P \text{ where } P \in P, \text{ iff } P \in \xi(s_i);$$

$$(S, s_i) \models \neg f_1 \text{ iff not } (S, s_i) \models f_1;$$

$$(S, s_i) \models f_1 \vee f_2 \text{ iff } (S, s_i) \models f_1 \text{ or } (S, s_i) \models f_2;$$

$$(S, s_i) \models \mathbf{X}f_1 \text{ iff } (S, s_{i+1}) \models f_1;$$

$$(S, s_i) \models f_1 \mathbf{U}f_2 \text{ iff there is a } j \geq i \text{ such that } (S, s_j) \models f_2 \text{ and for all } k \text{ with } i \leq k < j, (S, s_k) \models f_1;$$

$$(S, s_i) \models f_1 \mathbf{S}f_2 \text{ iff there is a } j \leq i \text{ such that } (S, s_j) \models f_2 \text{ and for all } k \text{ with } j < k \leq i, (S, s_k) \models f_1;$$

$$(S, s_i) \models N_j(f_1, \dots, f_n) \text{ (} 1 \leq j \leq m \text{) iff there is a finite or infinite string } a_{i_0} a_{i_1} a_{i_2} \dots \text{ generated by } G \text{ from } N_j \text{ such that, for all } k \geq 0, (S, s_{i+k}) \models f_{i_k}.$$

Example: Consider the grammar $N_1 \rightarrow a_1 a_2 N_1$. Let $f_1 = P$, $f_2 = \text{True}$, then $(S, s_i) \models N_1(f_1, f_2)$ iff the proposition P holds at all $i+2k$ ($k \geq 0$) positions of S since the only string that N_1 can generate is $a_1 a_2 a_1 a_2 \dots$, f_1 must be satisfied by S at all positions starting from s_i where a_1 occurs, namely all the positions of $i+2k$ ($k \geq 0$).

In the remainder of this paper, we always let G denote a regular right linear grammar with terminal symbols a_1, \dots, a_n and nonterminal symbols N_1, \dots, N_m .

2.2 Problems of propositional linear temporal logics

Satisfiability problem for linear temporal logic (SAT)

Given a formula $f \in L$, where L is a sublogic of $ETL(G)$, decide whether there is a structure $S = (s, \xi)$ such that $(S, s_0) \models f$.

Kripke Structure

A Kripke-structure K is a triple (N, R, η) , where N is a finite set of states (also called nodes), $R \subseteq N \times N$, and $\eta: N \rightarrow 2^P$. A path p in K is an infinite sequence (p_0, p_1, \dots) where $\forall i \geq 0, p_i \in N$, and $(p_i, p_{i+1}) \in R$. For a path p in a Kripke structure K , we let S_p denote the structure (s, ξ) where $\forall i \geq 0, \xi(s_i) = \eta(p_i)$.

Model checking Problem (MC)

Given a Kripke-structure $K=(N,R,\eta)$, a state $\delta \in N$, and a formula $f \in L$, where L is a sublogic of $ETL(G)$, decide whether there is a path p in K starting from δ such that $(S_p, s_0^p) \models f$.

Remark: The existential definition of Model checking problem here is the dual of the usual universal definition of model checking in verification. All the complexity results can be translated between the two formulations via duality.

Dual formulas

Let $f \in L$, where L is a sublogic of $ETL(G)$, the dual formula \bar{f} of f is defined by the following rules:

$$\overline{\bar{P}} = \neg P, \quad \overline{\neg P} = P;$$

$$\overline{\neg f_1} = \neg \bar{f}_1 \quad \text{if } f_1 \text{ isn't an atomic proposition};$$

$$\overline{Xf_1} = X \bar{f}_1;$$

$$\overline{f_1 Op f_2} = \bar{f}_1 Op \bar{f}_2, \quad \text{where } Op = \vee, U, S \text{ respectively};$$

$$\overline{N_j(f_1, \dots, f_n)} = N_j(\bar{f}_1, \dots, \bar{f}_n) \quad (1 \leq j \leq m).$$

It is easily seen that $\overline{\bar{f}} = f$.

Dual structures

Two structures $S=(s,\xi)$ and $T=(t,\pi)$ are called dual structures if for all $i \geq 0$,

$$\xi(s_i) \cap \pi(t_i) = \emptyset \quad \text{and} \quad \xi(s_i) \cup \pi(t_i) = P.$$

The dual structure of $S=(s,\xi)$ is denoted by $\bar{S}=(\bar{s},\bar{\xi})$.

Dual models

Given a formula $f \in L$, where L is a sublogic of $ETL(G)$, structures $S=(s,\xi)$ and $T=(t,\pi)$ are called dual models of f if $(S, s_0) \models f$, $(T, t_0) \models f$, and S, T are dual structures.

Example. Let $P=\{P_1, P_2\}$, $f=P_1UP_2$, $S=(s,\xi)$ be a structure with $\xi(s_0)=\{P_2\}$, $\xi(s_i)=\{P_1\}$ ($i \geq 1$), and $T=(t,\pi)$ be a structure with $\pi(t_0)=\{P_1\}$, $\pi(t_i)=\{P_2\}$ ($i \geq 1$), then S and T are dual models of f .

Dual paths in a Kripke-structure

Given a Kripke-structure $K=(N,R,\eta)$, two paths p and q are called dual paths if S_p and S_q are dual structures.

Dual models problem (DM)

Given a formula $f \in L$, where L is a sublogic of $ETL(G)$, decide whether f has dual models.

Determination of dual models in a Kripke-structure problem (KDM)

Given a Kripke-structure $K=(N,R,\eta)$, two states $\delta, \bar{\delta} \in N$, and a formula $f \in L$, where L is a sublogic of $ETL(G)$, decide whether there are dual paths p, q in K starting from $\delta, \bar{\delta}$ respectively such that $(S_p, s_0^p) \models f$ and $(S_q, s_0^q) \models f$.

Not-all-equal Satisfiability Problem for boolean formula^[11] (NAESAT)

Given a boolean formula $g=C_1 \wedge C_2 \wedge \dots \wedge C_m$ in 3-CNF where $C_i = l_{i_1} \vee l_{i_2} \vee l_{i_3}$ ($1 \leq i \leq m$), $l_{i_k} = x_j$ or $\neg x_j$ ($1 \leq k \leq 3$) for some j such that $1 \leq j \leq n$, x_1, \dots, x_n are the variables appearing in g , decide whether there exists an assignment $\eta: \{x_1, \dots, x_n\} \rightarrow \{\text{true}, \text{false}\}$ such that under this assignment three literals of each clause are neither all true nor all false.

It is evident that there exists an assignment $\eta: \{x_1, \dots, x_n\} \rightarrow \{\text{true}, \text{false}\}$ such that under this assignment three literals of each clause of g are neither all true nor all false iff there are two dual assignments η and $\bar{\eta}$ (namely $\bar{\eta}(x_i) = \text{true} \Leftrightarrow \eta(x_i) = \text{false}$ for all $1 \leq i \leq n$) such that g is true both under η and under $\bar{\eta}$.

3 Properties of Dual Models

Theorem 3.1. Let $f \in ETL(G)$, $S=(s, \xi)$ be a structure, then $(S, s_i) \models f$ iff $(\bar{S}, \bar{s}_i) \models \bar{f}$, where \bar{S} is the dual structure of S , and \bar{f} is the dual formula of f .

Proof: We prove the theorem by structural induction on f .

$f=P$ or $f=\neg P$: the two cases are trivial.

$f=\neg f_1$ where f_1 isn't an atomic proposition:

$(S, s_i) \models f$ iff not $(S, s_i) \models f_1$ iff

not $(\bar{S}, \bar{s}_i) \models \bar{f}_1$ (by induction hypothesis) iff

$(\bar{S}, \bar{s}_i) \models \neg \bar{f}_1$ iff $(\bar{S}, \bar{s}_i) \models \bar{f}$

$f=f_1 \vee f_2$:

$(S, s_i) \models f$ iff $((S, s_i) \models f_1$ or $(S, s_i) \models f_2)$ iff

$(\bar{S}, \bar{s}_i) \models \bar{f}_1$ or $(\bar{S}, \bar{s}_i) \models \bar{f}_2$ (by induction hypothesis) iff

$(\bar{S}, \bar{s}_i) \models \bar{f}_1 \vee \bar{f}_2$ iff $(\bar{S}, \bar{s}_i) \models \bar{f}_1 \vee \bar{f}_2$ iff $(\bar{S}, \bar{s}_i) \models \bar{f}$.

$f=Xf_1$: $(S, s_i) \models f$ iff $(S, s_{i+1}) \models f_1$ iff $(\bar{S}, \bar{s}_{i+1}) \models \bar{f}_1$ (by induction hypothesis) iff

$(\bar{S}, \bar{s}_i) \models X \bar{f}_1$ iff $(\bar{S}, \bar{s}_i) \models X \bar{f}_1$ iff $(\bar{S}, \bar{s}_i) \models \bar{f}$.

$f=f_1 U f_2$: $(S, s_i) \models f$ iff $((S, s_j) \models f_2$ for some $j \geq i$, and for all $k: i \leq k < j, (S, s_k) \models f_1)$ iff

$((\bar{S}, \bar{s}_j) \models \bar{f}_2$ for some $j \geq i$, and for all $k: i \leq k < j, (\bar{S}, \bar{s}_k) \models \bar{f}_1)$ (by induction hypothesis) iff

$(\bar{S}, \bar{s}_i) \models \bar{f}_1 U \bar{f}_2$ iff $(\bar{S}, \bar{s}_i) \models \bar{f}$

$f=f_1 S f_2$: similar to the case of $f=f_1 U f_2$.

$f=N_j f_1, \dots, f_n$ ($1 \leq j \leq m$): $(S, s_i) \models f$ iff

(there exists a finite or infinite string $a_{k_0}, a_{k_1}, a_{k_2}, \dots$ generated by G from N_j such that, for all $l \geq 0$,

$(S, s_{i+l}) \models f_{k_l}$) iff

(there exists a finite or infinite string $a_{k_0}, a_{k_1}, a_{k_2}, \dots$ generated by G from N_j such that, for all $l \geq 0$,

$(\bar{S}, \bar{s}_{i+l}) \models \bar{f}_{k_l}$) (by induction hypothesis) iff

$(\bar{S}, \bar{s}_i) \models N_j(\bar{f}_1, \dots, \bar{f}_n)$ iff $(\bar{S}, \bar{s}_i) \models \bar{f}$.

Corollary 3.2. Let $f \in L$, where L is a sublogic of $ETL(G)$, then f is satisfiable iff $f \vee \bar{f}$ has dual models.

Proof:

“Only if” part: Suppose that f is satisfiable.

There exists a structure $S=(s, \xi)$ such that $(S, s_0) \models f$, then according to Theorem 3.1, $(\bar{S}, \bar{s}_0) \models \bar{f}$. Obviously, S and \bar{S} are dual models of $f \vee \bar{f}$.

“If” part: Suppose that $f \vee \bar{f}$ has dual models, then there are dual structures $S=(s, \xi)$ and $\bar{S}=(\bar{s}, \bar{\xi})$ such that $(S, s_0) \models f \vee \bar{f}$ and $(\bar{S}, \bar{s}_0) \models f \vee \bar{f}$. Then $((S, s_0) \models f$ or $(S, s_0) \models \bar{f})$ and $((\bar{S}, \bar{s}_0) \models f$ or $(\bar{S}, \bar{s}_0) \models \bar{f})$. If $(S, s_0) \models f$ or $(\bar{S}, \bar{s}_0) \models f$, then we are done. Otherwise, we have $(S, s_0) \models \bar{f}$ and $(\bar{S}, \bar{s}_0) \models \bar{f}$, then according to Theorem 3.1, $(\bar{S}, \bar{s}_0) \models f$ and $(S, s_0) \models f$, f is satisfiable.

Corollary 3.3. Let $f \in L$, where L is a sublogic of $ETL(G)$, then f has dual models iff $f \wedge \bar{f}$ is satisfiable.

Proof:

“Only if” part: Suppose that f has dual models, then there exist dual structures $S=(s, \xi)$ and $\bar{S}=(\bar{s}, \bar{\xi})$ such that $(S, s_0) \models f$ and $(\bar{S}, \bar{s}_0) \models f$. According to Theorem 3.1, $(\bar{S}, \bar{s}_0) \models \bar{f}$. Thus $(\bar{S}, \bar{s}_0) \models f \wedge \bar{f}$, $f \wedge \bar{f}$ is satisfiable.

“If” part: Suppose that $f \wedge \bar{f}$ is satisfiable, then there exists a structure $S=(s,\xi)$ such that $(S,s_0) \models f \wedge \bar{f}$, namely $(S,s_0) \models f$ and $(S,s_0) \models \bar{f}$. According to Theorem 3.1, $(\bar{S},\bar{s}_0) \models f$, thus f has dual models.

4 Complexity of DM and KDM Problems

4.1 Complexity of DM problem

Theorem 4.1. The complexity of DM problem for propositional linear temporal logics is as follows:

- (i) DM is NP-complete for $L(F)$ and $\tilde{L}(F, X)$;
- (ii) DM is PSPACE-complete for $L(F, X), L(U), L(U, S, X), ETL(G)$.

Proof: (i) At first we show that DM is NP-hard for $L(F)$ and $\tilde{L}(F, X)$.

From Ref.[2], we know that SAT problems for $L(F)$ and $\tilde{L}(F, X)$ are NP-complete. Then if we can reduce SAT to DM for the two logics, we are done. As a matter of fact, according to Corollary 3.2, a formula $f \in L(F)$ ($f \in \tilde{L}(F, X)$ resp.) is satisfiable iff $f \vee \bar{f}$ (which is in $\tilde{L}(F, X)$ if $f \in \tilde{L}(F, X)$) has dual models, thus we have reduced SAT to DM for $L(F)$ and $\tilde{L}(F, X)$, and the reduction is obviously a polynomial-time reduction.

Now we show that DM is in NP by reducing DM to SAT for $L(F)$ and $\tilde{L}(F, X)$.

From Corollary 3.3, a formula $f \in L(F)$ ($f \in \tilde{L}(F, X)$ resp.) has dual models iff $f \wedge \bar{f}$ (which is in $\tilde{L}(F, X)$ if $f \in \tilde{L}(F, X)$) is satisfiable. Then we really have reduced DM to SAT for $L(F)$ and $\tilde{L}(F, X)$, and it is obvious that the reduction is a polynomial-time reduction. DM is in NP for $L(F)$ and $\tilde{L}(F, X)$, and thus DM is NP-complete for $L(F)$ and $\tilde{L}(F, X)$.

(ii) The proof is similar to (i). In Ref.[2], we have known that SAT is PSPACE-complete for $L(F, X), L(U), L(U, S, X), ETL(G)$, consequently, DM is PSPACE-complete for $L(F, X), L(U), L(U, S, X), ETL(G)$ as well.

4.2 Complexity of KDM problem

Theorem 4.2. The complexity of KDM problem for propositional linear temporal logics is as follows:

- (i) KDM is NP-complete for $L(F), \tilde{L}(F, X)$;
- (ii) KDM is PSPACE-complete for $L(F, X), L(U), L(U, S, X), ETL(G)$.

Theorem 4.2 is proved by the following two lemmas, namely Lemmas 4.3 and 4.4.

Lemma 4.3. KDM is NP-hard for $L(F), \tilde{L}(F, X)$ and PSPACE-hard for $L(F, X), L(U), L(U, S, X), ETL(G)$.

Proof: We reduce MC to KDM to show that KDM is NP-hard for $L(F), \tilde{L}(F, X)$ and PSPACE-hard for $L(F, X), L(U), L(U, S, X), ETL(G)$ since we know that MC is NP-complete for $L(F), \tilde{L}(F, X)$ and PSPACE-complete for $L(F, X), L(U), L(U, S, X), ETL(G)$ in Ref.[2].

Let P be a set of atomic propositions, $K=(N,R,\eta)$ be a Kripke-structure such that $\eta(x) \subseteq P$ for all $x \in N, \delta \in N, f \in L$

with all atomic propositions in P (where L denotes any of $L(F), \tilde{L}(F, X), L(F, X), L(U), L(U, S, X), ETL(G)$). Now we construct another Kripke-structure $K'=(N',R',\eta')$ and two states δ'_1, δ'_2 in N' and a formula $f' \in L$ such that there is a path p in K starting from δ such that $(S_p, s_0^p) \models f$ iff there are two dual paths p' and \bar{p}' starting from δ'_1, δ'_2 respectively such that $(S_{p'}, s_0^{p'}) \models f'$ and $(S_{\bar{p}'}, s_0^{\bar{p}'}) \models \bar{f}'$.

Let $P' = P \cup \{PI\}$. $K'=(N',R',\eta')$ is defined as follows (Fig.1):

- $N' = N \cup \{\bar{x} \mid x \in N\}$;
- $R' = R \cup \{(\bar{x}, \bar{y}) \mid (x, y) \in R\}$;
- $\eta'(x) = \eta(x) \cup \{PI\}$, $\eta'(\bar{x}) = P' - \eta'(x) = P - \eta(x)$.

And let $\delta'_1 = \delta, \delta'_2 = \bar{\delta}, f' = (f \wedge PI) \vee (\bar{f} \wedge \neg PI)$.

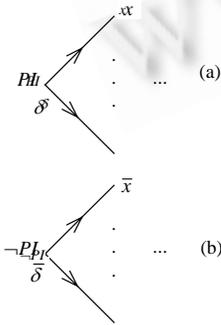


Fig.1

From the construction, it is not hard to see that there is a path p in Fig.1(a) starting from δ such that $(S_p, s_0^p) \models f$ iff there is a dual path \bar{p} of p in Fig.1(b) starting from $\bar{\delta}$ such that $(S_{\bar{p}}, s_0^{\bar{p}}) \models \bar{f}$. (*)

Now we prove that there is a path p in K starting from δ such that $(S_p, s_0^p) \models f$ iff there are two dual paths p' and \bar{p}' starting from δ'_1, δ'_2 respectively such that $(S_{p'}, s_0^{p'}) \models f'$ and $(S_{\bar{p}'}, s_0^{\bar{p}'}) \models f'$.

“Only if” part: Suppose that there is a path p in K starting from δ such that $(S_p, s_0^p) \models f$, then there exists a path p' in Fig.1(a) starting from δ'_1 (namely δ) such that $(S_{p'}, s_0^{p'}) \models f$, and consequently, $(S_{p'}, s_0^{p'}) \models f \wedge PI$, $(S_{p'}, s_0^{p'}) \models (f \wedge PI) \vee (\bar{f} \wedge \neg PI)$. According to (*), there is a dual path \bar{p}' of p' in Fig.1(b) starting from δ'_2 (namely $\bar{\delta}$) such that $(S_{\bar{p}'}, s_0^{\bar{p}'}) \models \bar{f}$, and consequently, $(S_{\bar{p}'}, s_0^{\bar{p}'}) \models \bar{f} \wedge \neg PI$, $(S_{\bar{p}'}, s_0^{\bar{p}'}) \models (f \wedge PI) \vee (\bar{f} \wedge \neg PI)$.

“If” part: Suppose that there are dual paths p' and \bar{p}' starting from δ'_1, δ'_2 respectively such that $(S_{p'}, s_0^{p'}) \models f'$ and $(S_{\bar{p}'}, s_0^{\bar{p}'}) \models f'$, then $(S_{p'}, s_0^{p'}) \models f \wedge PI$ and $(S_{\bar{p}'}, s_0^{\bar{p}'}) \models \bar{f} \wedge \neg PI$. Thus path p' is in part Fig.1(a), consequently, there is a path p in K starting from δ such that $(S_p, s_0^p) \models f$.

Thus we have reduced MC to KDM in polynomial time and consequently, KDM is NP-hard for $L(F), \tilde{L}(F, X)$ and is PSPACE-hard for $L(F, X), L(U), L(U, S, X), ETL(G)$.

Lemma 4.4. KDM is in NP for $L(F), \tilde{L}(F, X)$ and in PSPACE for $L(F, X), L(U), L(U, S, X), ETL(G)$.

Proof: We reduce KDM to MC for $L(F), \tilde{L}(F, X), L(F, X), L(U), L(U, S, X)$ and $ETL(G)$.

Let $K=(N, R, \eta)$ be a Kripke-structure with $\eta(x) \subseteq P$ for all $x \in N$, and $\delta_1, \delta_2 \in N; f \in L$ with all the atomic propositions in P (where L denotes any of $L(F), \tilde{L}(F, X), L(F, X), L(U), L(U, S, X)$ and $ETL(G)$).

By dualizing the assignments of atomic propositions of nodes, we get a Kripke-structure $\bar{K}=(\bar{N}, \bar{R}, \bar{\eta})$, where, $\bar{N}=\{\bar{x} | x \in N\}$, $\bar{R}=\{(\bar{x}, \bar{y}) | (x, y) \in R\}$ and $\bar{\eta}(\bar{x})=P-\eta(x)$.

Construct another Kripke-structure $K^*=(N^*, R^*, \eta^*)$ from K and \bar{K} as follows:

$$N^* = \{(s_1, \bar{s}_2) | s_1 \in N, \bar{s}_2 \in \bar{N}, \eta(s_1) = \bar{\eta}(\bar{s}_2)\};$$

R^* is defined by the rule: $((s_1, \bar{s}_2), (t_1, \bar{t}_2)) \in R^*$ iff $(s_1, t_1) \in R$ and $(\bar{s}_2, \bar{t}_2) \in \bar{R}$, where $(s_1, \bar{s}_2), (t_1, \bar{t}_2) \in N^*$;

$$\eta^*(s_1, \bar{s}_2) = \eta(s_1) \text{ for } (s_1, \bar{s}_2) \in N^*.$$

Now we prove that there are dual paths p, q in K starting from δ_1, δ_2 respectively such that $(S_p, s_0^p) \models f$ and $(S_q, s_0^q) \models \bar{f}$ iff $(\delta_1, \bar{\delta}_2) \in N^*$ and there is a path p^* in K^* starting from $(\delta_1, \bar{\delta}_2)$ such that $(S_{p^*}, s_0^{p^*}) \models f \wedge \bar{f}$.

“Only if” Part: Suppose that there are dual paths p, q in K starting from δ_1, δ_2 respectively such that $(S_p, s_0^p) \models f$ and $(S_q, s_0^q) \models \bar{f}$.

It is easy to see that there is a path \bar{q} in \bar{K} starting from $\bar{\delta}_2$ such that $\bar{\eta}(\bar{q}_i) = P - \eta(q_i)$ for all $i \geq 0$ and $(S_{\bar{q}}, s_0^{\bar{q}}) \models \bar{f}$ according to Theorem 3.1.

Then for all $i \geq 0$, $\bar{\eta}(\bar{q}_i) = P - \eta(q_i) = \eta(p_i)$, thus $(p_i, \bar{q}_i) \in N^*$. And for all $i \geq 0$, $(p_i, p_{i+1}) \in R$, $(\bar{q}_i, \bar{q}_{i+1}) \in \bar{R}$, thus $((p_i, \bar{q}_i), (p_{i+1}, \bar{q}_{i+1})) \in R^*$. Consequently, $p^* = (p_0, \bar{q}_0)(p_1, \bar{q}_1) \dots (p_i, \bar{q}_i) \dots$ is a path in K^* starting from $(\delta_1, \bar{\delta}_2)$ such that $(S_{p^*}, s_0^{p^*}) \models f$ and $(S_{p^*}, s_0^{p^*}) \models \bar{f}$ since $(S_p, s_0^p) \models f$ and $(S_{\bar{q}}, s_0^{\bar{q}}) \models \bar{f}$.

“If” Part: Suppose that $(\delta_1, \bar{\delta}_2) \in N^*$ and there is a path $p^* = (p_0, \bar{q}_0)(p_1, \bar{q}_1) \dots (p_i, \bar{q}_i) \dots$ in K^* starting from $(\delta_1, \bar{\delta}_2)$ such that $(S_{p^*}, s_0^{p^*}) \models f \wedge \bar{f}$, then for all $i \geq 0$, $\eta(p_i) = \bar{\eta}(\bar{q}_i)$, $(p_i, p_{i+1}) \in R$ and $(\bar{q}_i, \bar{q}_{i+1}) \in \bar{R}$. Thus $p = p_0 p_1 \dots p_i \dots$ is a path in K starting from δ_1 such that $(S_p, s_0^p) \models f$ and $\bar{q} = \bar{q}_0 \bar{q}_1 \dots \bar{q}_i \dots$ is a path in \bar{K} starting from $\bar{\delta}_2$ such that $(S_{\bar{q}}, s_0^{\bar{q}}) \models \bar{f}$, then there is a path q in K starting from δ_2 such that $\eta(q_i) = P - \bar{\eta}(\bar{q}_i) = P - \eta(p_i)$ for all $i \geq 0$, and $(S_q, s_0^q) \models \bar{f}$ according to Theorem 3.1. Now we get the desired dual paths p, q in K starting from

δ_1, δ_2 respectively.

Thus we have reduced KDM to MC and it is easy to show that the reduction is in polynomial time, then KDM is in NP for $L(F)$, $\tilde{L}(F, X)$ and in PSPACE for $L(F, X)$, $L(U)$, $L(U, S, X)$ and $ETL(G)$ since MC is shown to be NP-complete for $L(F)$, $\tilde{L}(F, X)$ and PSPACE-complete for $L(F, X)$, $L(U)$, $L(U, S, X)$ and $ETL(G)$ in Ref.[2].

5 Conclusions and Remarks

In this paper, we first defined the concept of dual models of propositional linear temporal logic formulas, and then investigated the complexity of dual models problem (DM) and the problem of determination of dual models in a Kripke-structure (KDM) for various propositional linear temporal logics. We proved that the DM and KDM for $L(F)$ and $\tilde{L}(F, X)$ are NP-complete, DM and KDM for $L(F, X)$, $L(U)$, $L(U, X)$, $L(U, S, X)$, $ETL(G)$ are PSPACE-complete (Table 1).

Table 1 The Complexity of DM and KDM

Logic	DM (dual models)	KDM (determination of dual models in Kripke-structure)
$L(F)$	NP-Complete	NP-Complete
$\tilde{L}(F, X)$		
$L(F, X)$	PSPACE-Complete	PSPACE-Complete
$L(U)$		
$L(U, X)$		
$L(U, S, X)$		
$ETL(G)$		

As a matter of fact, the reductions used in Theorem 4.1, Lemma 4.3 and Lemma 4.4 are general enough to determine the complexity of almost all propositional linear temporal logics defined so far only if the logic admits complete Boolean operators (since in the proof of Theorem 4.1, Lemma 4.2 and Lemma 4.3, “ \wedge ” and “ \vee ” operators are necessary). For instance, in Ref.[5], the complexity of SAT and MC problems has been investigated systematically for various fragments of propositional linear temporal logics by bounding the number of atomic propositions and the number of nesting of temporal operators used in temporal logic formulas. By applying Theorem 4.1, Lemmas 4.3 and 4.4, we can conclude that DM and KDM problems have exactly the same complexity as SAT and MC problems for those logics defined in Ref.[5].

It is also interesting to investigate further whether there are other variants of SAT for Boolean logic that can be generalized to propositional linear temporal logics.

References:

- [1] Pnueli A. The temporal logic of programs. In: Proc. of the 18th Annual Symp. on Foundations of Computer Science. New York: IEEE, 1977. 161–171.
- [2] Sistla AP, Clarke EM. The complexity of propositional linear temporal logics. Journal of ACM, 1985,32(3):733–749.
- [3] Emerson EA, Evangelist M, Srinivasan J. On the limits of efficient temporal decidability. In: Proc. of the LICS'90. IEEE, 1990. 464–475.
- [4] Chen CC, Lin IP. The computational complexity of satisfiability of temporal horn formulas in propositional linear-time temporal logic. Information Processing Letters, 1993,45(3):131–136.
- [5] Demri S, Schnoebelen PH. The complexity of propositional linear temporal logics in simple cases. In: Proc. of the STACS'98. LNCS 1373, Springer-Verlag, 1998. 61–72.
- [6] Kupferman O, Vardi MY. Relating linear and branching model checking. In Proc. of the IFIP Working Conf. on Programming Concepts and Methods (PROCOMET'98). 1998. 304–326.
- [7] Clarke EM, Emerson EA. Synthesis of synchronization skeletons for branching time temporal logic. In: Proc. of the Logic of Programs: Workshop. LNCS 131, Springer-Verlag, 1981. 52–71.

- [8] Manna Z, Wolper P. Synthesis of communicating processes from temporal logic specifications. ACM Trans. on Programming Languages and Systems, 1984,6(1):68–93.
- [9] Clarke EM, Emerson EA, Sistla AP. Automatic verification of finite-state concurrent systems using temporal logic specifications. ACM Trans. on Programming Languages and Systems, 1986,8(2):244–263.
- [10] Manna Z, Pnueli A. Verification of concurrent programs: A temporal proof system. In: Proc. of the 4th School on Advanced Programming. 1982.
- [11] Papadimitriou CH. Computational Complexity. Addison Wesley, 1993.



WU Zhi-Lin was born in 1980. He is a Ph.D. candidate at the Institute of Software, the Chinese Academy of Sciences. His current research areas are temporal logic and model checking.



ZHANG Wen-Hui was born in 1963. He is a professor at the Institute of Software, the Chinese Academy of Sciences. His research areas are formal methods and software reliability for developing high quality software.

www.jos.org.cn

www.jos.org.cn