Combining Unification and Rewriting in Proofs for Modal Logics with First-order Undefinable Frames

Shigeki Hagihara[1], Masahiko Tomoishi[1], Masaya Shimakawa[1], Naoki Yonezaki[2]

[1] Tokyo Institute of Technology, 2-12-1 Ookayama, Meguro-ku, Tokyo 152-8552 Japan.

[2] Tokyo Denki University, 2-1200 Muzai Gakuendai, Inzai-shi, Chiba 270-1382 Japan

Keywords: Modal logic; Proof method; Unification; Rewriting

Abstract. We provide a unification-based resolution method for basic modal logics. Because we use a clausal normal form that is quite similar to that in first-order logic, our method has good prospects for importing proof strategies for resolution methods from first-order logic. Furthermore, we show a solution for obtaining a resolution method for the modal logic KM, the frames of which are first-order and undefinable. It is impossible to make a unification rule for the modal logics of first-order undefinable frames in a similar way to that of basic modal logics. In this paper, we use a clausal rewriting rule for KM in addition to modal unification. We expect that this kind of adaptation can be applied to the construction of unification-based proof methods for other modal logics with first-order undefinable frames.

Introduction

Modal logics are used widely in various research fields, such as artificial intelligence (AI) and software verification. For example, in AI, the modal logics KT5 and KD45, called epistemic logics, are used for knowledge representation. In software verification, temporal logics such as LTL [1], CTL [2], and quantitative extension of LTL [3,4], which are extensions of the modal logics K4, KD4, and KT4, are used as specification languages for desirable properties of systems. Furthermore, the modal logics KT5 and KD45 are used in security protocol analysis [5–8]. In these fields, proof methods for modal logics play important roles in knowledge inference, software verification, and security analysis. Efficient proof methods are desirable in these fields.

In this paper, we define a resolution method for modal logic KM, a logic in which frames are first-order undefinable. Because the axiom M represents a time sequence model that will reach final states, this method can be used to prove dead lock-free systems.

First, we construct a resolution method for the basic modal logics frames that are restricted to first-order definable frames. Among various methods of proving modal formulae [9–15], unification-based proof methods [16,17] are efficient and have the ability to be adapted to various modal logics, because modal unification [18] absorbs differences in the modal logics. In previously reported proof methods for temporal logics [11,15], a clausal normal form was used; however, it did not reduce disjunction inside $\Box$ and conjunction inside $\Diamond$. In this paper, we introduce another type of clausal normal form. In it, each literal has a sequence of labeled modal operators as a prefix, and the labels correspond to Skolem function symbols in the first-order language used in specifying the semantics of modal logics. A clause is a disjunction of such prefixed literals. Our resolution method is a good prospect for introducing proof strategies for resolution methods for first-order logic, because our clausal normal form is similar to that used in the resolution method of first-order logic. Furthermore, it is easy to understand the flow of proof in our resolution method, because we do not have to reduce disjunction inside $\Box$ and conjunction inside $\Diamond$ in the middle of the proof, and the proof applies resolution rules alone.

Next, we extend this resolution method to deal with the modal logic KM. It is impossible to correspond labels with Skolem function symbols in the first-order language, because frame restriction is not first-order undefinable. To accommodate this, we use a rewriting rule based on axiom M in addition to modal unification.

Basic Modal Logics

There are various basic modal logics, as follows (see [19,20]):

- K, KD, KT, K4, KB, K5, KT4 (S4), KD4, KB4, KTB, KDB, K45, KT5 (S5), KD5, KD45.

In this section, we introduce the basic modal logics from [20]. Each has its own semantics. We introduce a common syntax and semantics for modal logics.

Syntax: Formulae in modal logics are defined inductively, as follows:

- Atomic propositions are formulae.
- $f \wedge g$, $f \vee g$, $\neg f$, $\Box f$, $\Diamond f$, $\bot$ are formulae, if f and g are formulae.

$\wedge$, $\vee$, and $\neg$ are the usual operators of 'classic' logic. $\bot$ is an atomic proposition representing falsity. $\Box$ and $\Diamond$ are modal operators, known as the necessity operator and the possibility operator, respectively. The modal logics K, KD, and KT are alethic logics. In these logics, $\Box f$ and $\Diamond f$ represent f necessarily holds and f possibly holds, respectively. The modal logics K4, KD4, and KT4 are bases of temporal logics. In these logics, $\Box f$ and $\Diamond f$ represent f always holds and f eventually holds, respectively. The modal logics KT5 and KD45 are epistemic logics. In these logics, $\Box f$ and $\Diamond f$ represent the statement that f holds is known and the statement that f does not hold is unknown, respectively.

Semantics: We give an interpretation to a formula, to define the semantics for the basic modal logics. A frame is a tuple $\langle W, R\rangle$ and a model is a triple $\langle W, R, V\rangle$, where W is a set of worlds, R is a binary relation on W (sometimes called a reachability relation), and V is an assignment that gives a set of worlds to a proposition symbol.
Formulae are interpreted by models. $M, w \vDash f$ denotes that a formula f is true at a world $w \in W$ in a model $M = \langle W, R, V\rangle$. The truth condition is defined as follows:

$$M, w \vDash p \Leftrightarrow w \in V(p)$$
$$M, w \vDash \bot \Leftrightarrow \bot$$
$$M, w \vDash \neg f \Leftrightarrow \neg(M, w \vDash \neg f)$$
$$M, w \vDash f \wedge g \Leftrightarrow (M, w \vDash f) \wedge (M, w \vDash g)$$
$$M, w \vDash f \vee g \Leftrightarrow (M, w \vDash f) \vee (M, w \vDash g)$$
$$M, w \vDash \Box f \Leftrightarrow \forall w' \in W(wRw' \rightarrow M, w' \vDash f)$$
$$M, w \vDash \Diamond f \Leftrightarrow \exists w' \in W(wRw' \wedge M, w' \vDash f)$$

The basic modal logics are classified by their own frame conditions. The frame conditions for the basic modal logic $KS_1 \ldots S_n$ is a conjunction of the conditions corresponding to $S_1, \ldots, S_n$, as listed in Table 1. For example, the frame conditions for KD4 are seriality and transitivity, and the frame conditions for KT5 are reflexivity and Euclidean property. If a frame $\langle W, R\rangle$ satisfies conditions of the modal logic $KS_1 \ldots S_n$, we say $\langle W, R\rangle$ is a $KS_1 \ldots S_n$-frame, and $\langle W, R, V\rangle$ is a $KS_1 \ldots S_n$-model.

A formula f is valid (unsatisfiable) in the class of $KS_1 \ldots S_n$-frames if for every $KS_1 \ldots S_n$-model $M = \langle W, R, V\rangle$ and for every world $w \in W$, $M, w \vDash f$ $(\neg(M, w \vDash f))$. A formula f is valid (with respect to being satisfiable, unsatisfiable) in the modal logic $KS_1 \ldots S_n$, if f is valid (with respect to being satisfiable, unsatisfiable) in the class of $KS_1 \ldots S_n$-frames.

Table 1. Axioms and conditions of reachability relations

Axioms	Conditions	
D	Serial	$\forall x \exists y\ xRy$
T	Reflexive	$\forall x\ xRx$
4	Transitive	$\forall xyz(xRy \wedge yRz \rightarrow xRz)$
B	Symmetric	$\forall xy(xRy \rightarrow yRx)$
5	Euclidean	$\forall xyz(xRy \wedge xRz \rightarrow yRz)$

Clausal Normal Form

In our resolution method, formulae are converted into a clausal normal form, which we introduce in this section. In our clausal normal form, each literal has a sequence of labeled modal operators as a prefix, and the clause is a disjunction of such prefixed literals.

We assume that all $\neg$ operators in a formula in the modal logic occur in front of proposition symbols. This restriction maintains generality.

We consider the first-order language $\mathfrak{L}$. In $\mathfrak{L}$, we have predicate $P(w)$, which has the same truth value as $w \in V(p)$ for each proposition p in the basic modal logic. For each formula f in the basic modal logic, we can consider the equivalent formula $\mathfrak{L}(f)$ in $\mathfrak{L}$. That is,

f is unsatisfiable in the class of $\mathrm{KS}_1 \ldots \mathrm{S}_n$-frames iff
'$\mathfrak{L}(f) \wedge$ the frame conditions for $\mathrm{KS}_1 \ldots \mathrm{S}_n$' is unsatisfiable in the first-order logic.

Here, we label each occurrence of $\Box$ and $\Diamond$ in a formula with a Skolem function symbol, which occurs in the Skolemized formula of $\mathfrak{L}(f) \wedge$ the frame conditions for $\mathrm{KS}_1 \ldots \mathrm{S}_n$.

Example 1 Let f be $\Box\Diamond\Diamond p$. Then, $\mathfrak{L}(f)$ is

$$\forall x(wRx \to \exists y\left(xRy \wedge \exists z\big(yRz \wedge P(z)\big)\right))$$

Hence, the Skolemized formula of $\mathfrak{L}(f)$ is

$$wRx \to (xRa(x) \wedge \left(a(x)Rb(x) \wedge P\big(b(x)\big)\right))$$

Thus, the labeled formula of f is

$$\Box_x \Diamond_a \Diamond_b p.$$

Now, we consider the correspondence between a labeled formula f^* and the Skolemized formula in $\mathfrak{L}$. For $\Diamond_a(p \wedge q)$, the Skolemized formula is $wRa(w) \wedge P(a(w)) \wedge Q(a(w))$. For $\Diamond_a p \wedge \Diamond_a q$, the Skolemized formula is $wRa(w) \wedge P(a(w)) \wedge wRa(w) \wedge Q(a(w))$. That is, $\Diamond_a(p \wedge q)$ has the equivalent satisfiability of $\Diamond_a p \wedge \Diamond_a q$. Similarly, for $\Box_x(p \vee q)$, the Skolemized formula is $wRx \to (P(x) \vee Q(x))$. For $\Box_x p \vee \Box_x q$, the Skolemized formula is $(wRx \to P(x)) \vee (wRx \to Q(x))$. Thus, $\Box_x(p \vee q)$ has the equivalent satisfiability of $\Box_x p \vee \Box_x q$. These results mean that in addition to the usual distribution rules $\Box(f \wedge g) \Rightarrow \Box f \wedge \Box g$ and $\Diamond(f \vee g) \Rightarrow \Diamond f \vee \Diamond g$, we can use the following distribution rules due to the labeling.

$$\Box_x(f \vee g) \Rightarrow \Box_x f \vee \Box_x g$$
$$\Diamond_a(f \wedge g) \Rightarrow \Diamond_a f \wedge \Diamond_a g$$

Using these rules, we can translate a formula f into clausal normal form f^c, where each literal has a sequence of labeled modal operators as a prefix, and the clause is a disjunction of such prefixed literals.

Example 2 Let f be as follows.

$$f\colon\ \Diamond p \wedge \Box(\neg p \vee \Diamond q) \wedge \Box\neg q$$

The labeled formula f^* and the clausal normal form f^c are as follows:

$$f^*\colon\ \Diamond_a p \wedge \Box_x(\neg p \vee \Diamond_b q) \wedge \Box_y \neg q$$

$$f^c\colon\ \Diamond_a p \wedge (\Box_x \neg p \vee \Box_x \Diamond_b q) \wedge \Box_y \neg q$$

Unification-based Resolution Method for Basic Modal Logics

In this section, we introduce a unification-based resolution method for the basic modal logics. The resolution method is a refutation system. First, we transform a formula f to f^c. Then, we apply the following resolution rules to f^c. We say f or f^c is refutable if the empty clause $\bot$ is derived from f^c.

$$\text{rule1} \qquad \frac{\alpha L \vee \Gamma \quad \beta \bar{L} \vee \Gamma'}{(\alpha \bot \vee \Gamma \vee \Gamma')^{\sigma(\alpha,\beta)}}$$

$$\text{rule2} \qquad \frac{\alpha\gamma L \vee \Gamma \quad \beta\delta L' \vee \Gamma'}{(\alpha\gamma L \vee \Gamma \vee \Gamma')^{\sigma(\alpha,\beta)}}$$

$$\text{rule3} \qquad \frac{\alpha \bot \vee \Gamma}{\Gamma} \qquad \text{(if there is no } \Box \text{ in } \alpha\text{)}$$

where L, L' and $\bar{L}$ are literals; L and $\bar{L}$ are complementary literals; α, β, γ and δ are sequences of modal operators associated with labels; $\sigma(\alpha,\beta)$ is a substitution that unifies α and β; and $(\alpha \bot \vee \Gamma \vee \Gamma')^{\sigma(\alpha,\beta)}$ and $(\alpha\gamma L \vee \Gamma \vee \Gamma')^{\sigma(\alpha,\beta)}$ are the formulae obtained by replacing modal operators in $(\alpha \bot \vee \Gamma \vee \Gamma')$ and $(\alpha\gamma L \vee \Gamma \vee \Gamma')$ with the substitution $\sigma(\alpha,\beta)$, respectively. For the modal logic $KS_1 \ldots S_n$, each substitution should consist of the assignments corresponding to K, S_1, ..., S_n, as listed in Table 2. For example, in unification in KT4, assignments of the form $\{\Box,\Diamond\}/\Box$, $\emptyset/\Box$, $\{\Box,\Diamond\}^+/\Box$ are allowed. $\Diamond_f$ is a special constant-labeled modal operator $\Diamond$. The symbol + represents transitive closure. If the same variable-labels appear in different clauses, they are managed as different variable labels. Resolution rules 1 and 3 are usual rules. Rule 2 is used for replacing α with $\sigma(\alpha,\beta)$.

Table 2. Assignments for modal logics

Axioms	Type of assignments
K	$\{\Box,\Diamond\}/\Box$
D	$\Diamond_f/\Box$
T	$\emptyset/\Box$
4	$\{\Box,\Diamond\}^+/\Box$
B	$\emptyset/\Diamond\Box$
5	$\{\Box,\Diamond\}^+/\{\Box,\Diamond\}^+\Box$

Theorem 1 If a labeled formula f^c is refutable by the resolution method for the modal logic $KS_1 \ldots S_n$, f is unsatisfiable in the modal logic $KS_1 \ldots S_n$.

Example 3 A refutation of the following formula f in K4 is as follows:

$$f\text{:}\ \Diamond p \wedge \Box(\neg p \vee \Diamond q) \wedge \Box\neg q$$

As shown in Example 2, the clausal normal form f^c is as follows:

$$f^c\text{:}\ \Diamond_a p \wedge (\Box_x \neg p \vee \Box_x \Diamond_b q) \wedge \Box_y \neg q$$

Figure 1 shows a refutation of f^c. This means, $\Diamond p \wedge \Box(\neg p \vee \Diamond q) \wedge \Box\neg q$ is unsatisfiable in the modal logic K4.

$$\dfrac{\dfrac{\dfrac{\dfrac{\Diamond_a p \qquad \Box_x \neg p \vee \Box_x \Diamond_b q}{\Diamond_a \bot \vee \Diamond_a \Diamond_b q}\ \mathit{rule1}, \Diamond_a/\Box_x}{\Diamond_a \Diamond_b q}\ \mathit{rule3} \qquad \Box_y \neg q}{\Diamond_a \Diamond_b \bot}\ \mathit{rule1}, \Diamond_a\Diamond_b/\Box_y}{\bot}\ \mathit{rule3}$$

Figure 1. A refutation of $\Diamond p \wedge \Box(\neg p \vee \Diamond q) \wedge \Box\neg q$.

Unification-based resolution method for KM

In this section, we describe a solution for obtaining a resolution method for the modal logic KM, the frames of which are first-order and undefinable.

The axiomatic system of KM is the system obtained by adding the McKinsey axiom M: $\Box\Diamond A \rightarrow \Diamond\Box A$ to the axiomatic system for K. The frame condition for KM is not first-order definable [21]. Thus, it is impossible to define a unification-based resolution method for KM in a similar way as that for basic modal logics. We expect a pattern of unification from axiom M.

Because the negation of axiom M is $\Box\Diamond p \wedge \Box\Diamond\neg p$, a candidate for assignment may be $\Box_x\Diamond_a/\Box_y\Diamond_b$. However, because the frame condition for KM is not clarified, it is difficult to justify the candidate $\Box_x\Diamond_a/\Box_y\Diamond_b$. Hence, we adopt the addition of new clauses using rewriting, in addition to adaptation by unification. Axiom M: $\Box\Diamond A \rightarrow \Diamond\Box A$ can be considered the clause rewriting rule $\Box\Diamond A \Rightarrow \Diamond\Box A$. For $\Box_x\Diamond_a p \wedge \Box_y\Diamond_b\neg p$, we add the new clauses $\Diamond_{a'}\Box_{x'}p$ and $\Diamond_{b'}\Box_{y'}\neg p$. This makes refutation possible by the unification $\Diamond_{a'}\Box_{x'}p$ and $\Box_y\Diamond_b\neg p$ using the assignment $\{\Diamond_{a'}/\Box_y, \Diamond_b/\Box_{x'}\}$.

Related works

Resolution methods using a translation from a modal formula to a formula of clausal normal form of predicate logic were proposed in [13] and [14]. They are advantageous in making full use of proof strategies with resolution methods of predicate logic. They can adapt to modal logics with first-order definable frames. However, they cannot deal with KM, because their frame conditions are not first-order definable.

Proof methods for modal logics with first-order undefinable frames were suggested in [22] and [23]. The method proposed in [22] uses a combination of Hilbert-style reasoning and semantic reasoning. Our approach is similar for adaptation to KM. However, the method proposed in [23] uses translation from a modal formula into a formula of set theory. For adapting to KM, it would be necessary to translate the frame condition for KM into a formula in set theory.

Conclusions

We described unification-based resolution methods for basic modal logics. Because our clausal normal form is quite similar to that in first-order logic, we can import proof strategies that have been studied extensively in proof methods in first-order logic. We discussed a solution for obtaining a resolution method for the modal logic KM, the frames of which are first-order and undefinable. There are several axioms, such as N1, that characterize first-order undefinable frames. We expect that this kind of adaptation to KM can be applied to the construction of unification-based proof methods for other modal logics with first-order undefinable frames.

In addition, future research will include more practical applications of unification and rewriting in the proof method of the modal logic. We have proposed a practical application of a proof method for LTL, which is considered an extension of modal logic, in security analyses [24,25], bioinformatics [26–28], system verification [29–33], and system synthesis [34,35]. We will adapt modal unification and rewriting to these applications.

References

[1] Amir Pnueli. The temporal semantics of concurrent programs. Theoretical Computer Science, 13:45-60, 1981. doi:10.1016/0304-3975(81)90110-9

[2] Mordechai Ben-Ari, Zohar Manna, and Amir Pnueli. The temporal logic of branching time. 8th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '81, pages 164-176, New York, NY, USA, 1981. ACM. doi:10.1145/567532.567551

[3] Takashi Tomita, Shigeki Hagihara, and Naoki Yonezaki. A probabilistic temporal logic with frequency operators and its model checking. 13th International Workshop on Verification of Infinite-State Systems (INFINITY 2011), volume 73 of EPTCS, pages 79-93, 2011. doi:10.4204/EPTCS.73.9

[4] Takashi Tomita, Shin Hiura, Shigeki Hagihara, and Naoki Yonezaki. A temporal logic with mean-payoff constraints. 14th International Conference on Formal Engineering Methods: Formal Methods and Software Engineering, ICFEM'12, pages 249-265, Berlin, Heidelberg, 2012. Springer-Verlag. doi:10.1007/978-3-642-34281-3_19

[5] J.Y. Halpern and K.R. O'Neill. Anonymity and information hiding in multiagent systems. 16th IEEE Computer Security Foundations Workshop 2003, pages 75 - 88, 2003. doi:10.1109/CSFW.2003.1212706

[6] Flavio D. Garcia, Ichiro Hasuo, Wolter Pieters, and Peter van Rossum. Provable anonymity. In FMSE '05: the 2005 ACM workshop on Formal methods in security engineering, pages 63-72, New York, NY, USA, 2005. ACM. doi:10.1145/1103576.1103585

[7] Shigeki Hagihara, Hiroaki Oguro, and Naoki Yonezaki. Kripke semantics for epistemic logic of relational information between ciphertexts. Philippine Computing Journal, 7(2):23-32, Dec. 2012.

[8] Shigeki Hagihara, Hiroaki Oguro, and Naoki Yonezaki. Completeness of a deduction system for relational information between ciphertexts based on probabilistic computational semantics. Theory and Practice of Computation, volume 5 of Proceedings in Information and Communications Technology, pages 116-132. Springer, 2012. doi:10.1007/978-4-431-54106-6_10

[9] Melvin C. Fitting. Proof Methods for Modal and Intutionistic Logics. Reidel, Dordrecht, 1983.

[10] Annie Foret. Rewrite rule systems for modal propositional logic. The Journal of Logic Programming, 12:281-298, 1992. doi:10.1016/0743-1066(92)90028-2

[11] P. Enjalbert and L. Farinas del Cerro. Modal resolution in clausal form. Theoretical Computer Science, 65(1):1-33, 1989. doi:10.1016/0304-3975(89)90137-0

[12] Lincoln A. Wallen. Automated proof search in non-classical logics. MIT Press, 1990.

[13] H. J. Ohlbach. A resolution calculus for modal logics. Automated Deduction - CADE-9, volume 310 of Lecture Notes in Computer Science, pages 500-516. Springer, 1988. doi: 10.1007/BFb0012852

[14] Andreas Nonnengart. Resolution-based calculi for modal and temporal logics. Automated Deduction - CADE-13, volume 1104 of Lecture Notes in Artificial Intelligence, pages 598-612. Springer, 1996. doi:10.1007/3-540-61511-3_116

[15] Martín Abadi and Zohar Manna. Nonclausal deduction in first-order temporal logic. J. ACM, 37(2):279-317, April 1990. doi:10.1145/77600.77617

[16] Naoki Yonezaki and Takashi Hayama. Self-substitution in modal unification. Information Modeling and Knowledge Bases IV, pages 180-195. IOS Press, 1993.

[17] Shigeki Hagihara and Naoki Yonezaki. Resolution method for modal logic with well-founded frames. Computer Science Logic (CSL'99), volume 1683 of Lecture Notes in Computer Science, pages 277-291. Springer, 1999. doi:10.1007/3-540-48168-0_20

[18] Jens Otten and Christoph Kreitz. T-string unification: Unifying prefixes in non-classical proof methods. Theorem Proving with Analytic Tableaux and Related Methods, volume 1071 of Lecture Notes in Artificial Intelligence, pages 244-260. Springer, 1996. doi:10.1007/3-540-61208-4_16

[19] G. E. Hughes and M. J. Cresswell. A New Inroduction to Modal Logic. Routledge, 1996.

[20] Rajeev Gore. Technical report tr-arp-15-95. Technical report, Research School of Information Sciences and Engineering and Centre for Information Science Research Australian National University, http://arp.anu.edu.au/, 1995.

[21] Robert Goldblatt. Mathematics of Modality, volume 43 of CSLI Lecture Notes, chapter 10, pages 231-241. CSLI Publications, 1993.

[22] H. J. Ohlbach. Combining hilbert style and semantic reasoning in a resolution framework. Automated Deduction - CADE-15, volume 1421 of Lecture Notes in Artificial Intelligence, pages 205-219, 1998. doi:10.1007/BFb0054261

[23] G. D'Agostino, A. Montanari, and A. Policriti. A set-theoretic translation method for polymodal logics. Journal of Automated Reasoning, 15(3):317-337, 1995. doi:10.1007/BF00881803

[24] Ashraf Bhery, Shigeki Hagihara, and Naoki Yonezaki. A formal system for analysis of cryptographic encryption and their security properties. International Symposium on Software Security 2003, Software Security - Theories and Systems, volume 3233 of Lecture Notes in Computer Science, pages 87-112, 2004. doi:10.1007/978-3-540-37621-7_5

[25] Ashraf Bhery, Shigeki Hagihara, and Naoki Yonezaki. A formal analysis of symmetric encryption and keyed and keyed hash function. 46th IEEE International Midwest Symposium on Circuits & Systems 2003, 2003. doi:10.1109/MWSCAS.2003.1562401

[26] Sohei Ito, Takuma Ichinose, Masaya Shimakawa, Naoko Izumi, Shigeki Hagihara, and Naoki Yonezaki. Modular analysis of gene networks by linear temporal logic. Journal of Integrative Bioinformatics (JIB), 10(2)::216, 2013. doi:10.2390/biecoll-jib-2013-216

[27] Sohei Ito, Takuma Ichinose, Masaya Shimakawa, Naoko Izumi, Shigeki Hagihara, and Naoki Yonezaki. Qualitative analysis of gene regulatory networks using network motifs. International Conference on Bioinformatics Models, Methods and Algorithms (BIOSTEC 2013), pages 15-24, 2013. doi:10.5220/0004188400150024

[28] Sohei Ito, Takuma Ichinose, Masaya Shimakawa, Naoko Izumi, Shigeki Hagihara, and Naoki Yonezaki. Qualitative analysis of gene regulatory networks by temporal logic. Theoretical Computer Science, 594(23):151-179, August 2015. doi:10.1016/j.tcs.2015.06.017

[29] Shigeki Hagihara, Yusuke Kitamura, Masaya Shimakawa, and Naoki Yonezaki. Extracting environmental constraints to make reactive system specifications realizable. 16th Asia-Pacific Software Engineering Conference, APSEC '09, pages 61-68, Washington, DC, USA, 2009. IEEE Computer Society. doi:10.1109/APSEC.2009.70

[30] Shigeki Hagihara, Naoki Egawa, Masaya Shimakawa, and Naoki Yonezaki. Minimal strongly unsatisfiable subsets of reactive system specifications. 29th ACM/IEEE International Conference on Automated Software Engineering, ASE '14, pages 629-634, New York, NY, USA, 2014. ACM. doi:10.1145/2642937.2642968

[31] Shigeki Hagihara and Naoki Yonezaki. Completeness of verification methods for approaching to realizable reactive specifications. 1st Asian Working Conference on Verified Software AWCVS'06, volume 348 of UNU-IIST Technical Report, pages 242-257, 2006.

[32] Masaya Shimakawa, Shigeki Hagihara, and Naoki Yonezaki. Complexity of strong satisfiability problems for reactive system specifications. IEICE Transactions on Information and Systems, E96-D(10):2187-2193, Oct. 2013. doi:10.1587/transinf.E96.D.2187

[33] Masaya Shimakawa, Shigeki Hagihara, and Naoki Yonezaki. Bounded strong satisfiability checking of reactive system specifications. IEICE TRANSACTIONS on Information and Systems, 97(7):1746-1755, 2014. doi:10.1587/transinf.E97.D.1746

[34] Shigeki Hagihara, Atsushi Ueno, Takashi Tomita, Masaya Shimakawa, and Naoki Yonezaki. Simple synthesis of reactive systems with tolerance for unexpected environmental behavior. 4th FME Workshop on Formal Methods in Software Engineering, pages 15-21. ACM, 2016. doi:10.1145/2897667.2897672

[35] Takashi Tomita, Atsushi Ueno, Masaya Shimakawa, Shigeki Hagihara, and Naoki Yonezaki. Safraless LTL synthesis considering maximal realizability. Acta Informatica, pages 1-38, 2016. doi:10.1007/s00236-016-0280-3