Liet. mat. rink. LMD darbai, **48/49**, 2008, 269–274

Restrictions for loop-check in sequent calculus for temporal logic

Adomas BIRŠTUNAS (VU)
e-mail: adomas.birstunas@mif.vu.lt

Abstract. In this paper, we present sequent calculus for linear temporal logic. This sequent calculus uses efficient loop-check techinque. We prove that we can use not all but only several special sequents from the derivation tree for the loop-check. We use indexes to discover these special sequents in the sequent calculus. These restrictions let us to get efficient decision procedure based on introduced sequent calculus.

Keywords: sequent calculus, temporal logic, efficient loop-check.

1. Introduction

Usual sequent calculi with cut rule are practically unusable in automated environment. So, sequent calculi with analytic cut, or infinitary rule, or some kind of the loop-check must be used to get a decision procedure. Unfortunately, such a sequent calculi are inefficient and need additional modifications to get more or less usable decision procedure. One of the possibilities is constructing sequent calculi with an efficient loop-check (as it is done for some modal logics in [3]), or (if it is only possible) loop-check free calculi (as it is done for $KD45$ logic in [1]). The same situation holds for linear temporal logic.

There is known sequent calculus for linear temporal logic which is cut free and invariant free sequent calculus ([2]), but we need to use loop-check to get decidability and it uses *until* temporal operator. In [2], you can also find a survey about other known calculi for linear temporal logic. In this paper, we suggest some restrictions to the loop-check for linear temporal logic. We concentrate on the linear temporal logic with 2 modal operators: *allways* ($\Box$) and *next* ($\circ$). Restrictions allows us to use only special sequents in a loop-check. So, we got a sequent calculi with an efficient loop check.

There is known sequent calculus for linear temporal logic, which uses loop-check. Sequent calculus has logical rules and the following rules:

$$\frac{\Gamma\to\Delta, A \qquad \Gamma\to\Delta, \circ\Box A}{\Gamma\to\Delta, \Box A}\ (\to\Box) \qquad \frac{\Gamma\to\Delta}{\circ\Gamma\to\circ\Delta}\ (\circ),$$

$$\frac{A, \circ\Box A, \Gamma\to\Delta}{\Box A, \Gamma\to\Delta}\ (\Box\to) \qquad \frac{\Gamma\to\Delta}{\Gamma,\Gamma'\to\Delta,\Delta'}\ (Weak).$$

DEFINITION 1. We say that sequent S is an ancestor of the sequent S' in the derivation tree, if there exist sequence of the sequents $S_1 = S, S_2, \ldots, S_n = S'$ that for every

$i = 1, 2, \dots (n-1)$, sequent S_i is a conclusion and sequent S_{i+1} is a premise of some rule application.

DEFINITION 2. We say, that we have a loop $S \rightsquigarrow S'$ in the sequent derivation tree if the following conditions are true:
– sequent S is an ancestor of the sequent S';
– S' may be obtained from S by rule $(Weak)$ application: $\frac{S}{S'} \quad (Weak)$.

DEFINITION 3. We say, that sequent S' is a *loop-axiom* if there exists sequent S satisfying the following conditions:
– $S \rightsquigarrow S'$ is a loop in the derivation tree;
– between sequents S and S', there exists such a rule $(\rightarrow \Box)$ application, that its right premise is sequent S' ancestor.

DEFINITION 4. Sequent calculus for linear temporal logic with inference rules $(\rightarrow \vee)$, $(\vee \rightarrow)$, $(\rightarrow \&)$, $(\& \rightarrow)$, $(\rightarrow \neg)$, $(\neg \rightarrow)$, $(Weak)$, $(\rightarrow \Box)$, $(\Box \rightarrow)$, $(\circ)$, with a loop-axiom and with an axiom $\Gamma, \phi \rightarrow \Delta, \phi$ we call sequent calculus G_1LTL.

2. Restrictions for the loop-check

According sequent calculus G_1LTL, we can make a decision procedure (similar to the decision procedure described in [5]). Unfortunatelly, this procedure must use loop-check technique to detect non-derivable sequent, or to detect loop-axiom. In other words, we have to deal with two types of the loops. One is a loop-axiom (Definition 3) which may lead initial sequent to be derivable. For this type of the loop we use term loop-axiom. Other is a simple loop, which is not a loop-axiom, and leads initial sequent to be non derivable. For this type of the loop we use term 'nonderivable' loop. We use term loop to denote both types of the loop. All restrictions for loop check are applied for both types of the loop.

Now we introduce sequent calculus G_2LTL which uses only invertable or semi invertable rules. To get such a sequent calculus, we use primary sequents.

DEFINITION 5. Sequent S is *primary* if S has the shape $\Sigma, \circ\Gamma \rightarrow \Pi, \circ\Delta$ and Σ, Π contains only propositional variables, $\Sigma \cap \Pi = \emptyset$.

DEFINITION 6. Sequent calculus for linear temporal logic with inference rules $(\rightarrow \vee)$, $(\vee \rightarrow)$, $(\rightarrow \&)$, $(\& \rightarrow)$, $(\rightarrow \neg)$, $(\neg \rightarrow)$, $(\rightarrow \Box)$, $(\Box \rightarrow)$, $(\circ_p)$, with a loop-axiom and with an axiom $\Gamma, \phi \rightarrow \Delta, \phi$ we call sequent calculus G_2LTL.

$$\frac{\Gamma \rightarrow \Delta}{\Sigma, \circ\Gamma \rightarrow \Pi, \circ\Delta} \quad (\circ_p),$$

Σ, Π contains only propositional variables and $\Sigma \cap \Pi = \emptyset$.

Simple speaking, rule $(\circ_p)$ may be applied only for primary sequent.

LEMMA 1. *Sequent S is derivable in sequent calculus G_1LTL if and only if sequent S is derivable in sequent calculus G_2LTL.*

The proof is omitted because of the lack of the space.

Now we introduce sequent calculus G_3LTL wich uses loop-check only for sequents those are some premises of the rule $(\circ_p)$ application only. This modification reduces the number of the checked sequents in the derivation tree.

DEFINITION 7. Loop $S \rightsquigarrow S'$ is a $\circ$-loop if there exist primary sequents S_1, S_1' in the derivation tree, and S_1 is obtained from the sequent S, and S_1' is obtained from the sequent S' by rule $(\circ_p)$ applications (i.e., $\frac{S}{S_1}(\circ_p)$, $\frac{S'}{S_1'}(\circ_p)$).

DEFINITION 8. We say, that sequent S' is a $\circ$-*loop-axiom* if there exists sequent S satisfying the following conditions:

– $S \rightsquigarrow S'$ is a $\circ$-loop in the derivation tree;

– $S \rightsquigarrow S'$ is a loop-axiom.

DEFINITION 9. Sequent calculus for linear temporal logic with inference rules $(\rightarrow \vee)$, $(\vee \rightarrow)$, $(\rightarrow \&)$, $(\& \rightarrow)$, $(\rightarrow \neg)$, $(\neg \rightarrow)$, $(\rightarrow \Box)$, $(\Box \rightarrow)$, $(\circ_p)$, with a $\circ$-loop-axiom and with an axiom $\Gamma, \phi \rightarrow \Delta, \phi$ we call sequent calculus G_3LTL.

LEMMA 2. *Sequent S is derivable in sequent calculus G_2LTL if and only if sequent S is derivable in sequent calculus G_3LTL.*

The proof is omitted because of the lack of the space.

Now we prove some lemmas to introduce main restrictions to the loop-check used. First, we define subformulas and prove some features for them.

DEFINITION 10. We write $F \subseteq_{sf} G$ to define that F is subformula of G.

We use extended term subformula: if we have formula of the shape $\Box F$, then we say, that formula $\circ\Box F$ is also subformula of $\Box F$ ($\circ\Box F \subseteq_{sf} \Box F$).

DEFINITION 11. Formula F is *proper subformula* of G, if F is subformula of G and formula G length is greater then formula F length.

We write $F \subset_{sf} G$ to define that F is proper subformula of G.

DEFINITION 12. Formula F in sequent S is *ground* if for ever formula $G \in S$, formula F is not a proper subformula of G ($F \not\subset_{sf} G$).

LEMMA 3. *If $F \subset_{sf} G$, and $G \subseteq_{sf} H$, then $F \subset_{sf} H$ or $G = \circ H = \circ F = \circ\Box A$.*

Proof. If $G \subseteq_{sf} H$, then a) $G \subset_{sf} H$, or b) $G = H$, or c) $G = \circ H = \circ\Box A$.

In the cases a) and b), it is evident, that $F \subset_{sf} H$. In the case c), we just have $F \subseteq_{sf} H$. If H lenght greater then F length, then $F \subset_{sf} H$. If H and F lengths are equal then $F = H$ and $G = \circ H = \circ F = \circ\Box A$. If F lenght is greater then H length, then $F = \circ H$, and $G = \circ H = F$ – contradiction for $F \subset_{sf} G$.

LEMMA 4. *If $S \rightsquigarrow S'$ is a $\circ$-loop then, in any sequent $T \in S \rightsquigarrow S'$, any ground formula $F \in T$ has the shape $\Box A$ or $\circ\Box A$.*

Proof. Since $S \rightsquigarrow S'$ is a $\circ$-loop, there exists such a sequent S'_1, that S'_1 is obtained from the sequent S' by rule $(\circ_p)$ application $(\frac{S'}{S'_1}(\circ_p))$.

There exists ground formula $G \in S$, that $F \subseteq_{sf} G$. $G \in S'$ and $\circ G \in S'_1$, because $S \rightsquigarrow S'$ is a $\circ$-loop. There exists ground formula $H \in T$, that $\circ G \subseteq_{sf} H$.

We have, that $F \not\subset_{sf} H$ (because F is ground formula in T) and $F \subseteq_{sf} H$ ($F \subseteq_{sf} G$, $G \subset_{sf} \circ G$, $\circ G \subseteq_{sf} H$). Therefore, $F = \circ H = \circ\Box A$ (this case satisfies lemma) or $F = H$.

If $F = H$, then $G \subset_{sf} \circ G$ and $\circ G \subseteq_{sf} F (= H)$. According to Lemma 3, $G \subset_{sf} F$, or $\circ G = \circ F = \circ\Box A$, and $F = \Box A$ (satisfies lemma).

Since $G \subset_{sf} F$ and $F \subseteq_{sf} G$, then, according to Lemma 3, $G \subset_{sf} G$ (we get a contradiction), or $F = \circ G = \circ\Box A$ (satisfies lemma).

LEMMA 5. *Suppose, that $S \rightsquigarrow S'$ is a $\circ$-loop in the derivation tree constructed according sequent calculus G_3LTL. If F is ground formula in sequent S or in sequent S', then formula F or $\circ F$ is ground in all sequents in a $\circ$-loop $S \rightsquigarrow S'$.*

Proof. Every rule premise contains only subformulas of the rule conclusion.

Case 1) Ground formula $F \in S$. Then $F \in S'$, because $S \rightsquigarrow S'$ is a $\circ$-loop. Suppose, that formula F is not ground on some sequent T inside the $\circ$-loop $S \rightsquigarrow S'$. So, there exist ground formula $G \in T$, that $F \subset_{sf} G$.

There exist ground formula $H \in S$, that $G \subseteq_{sf} H$. According to Lemma 3, $F \subset_{sf} H \in S$ (contradiction for F being ground in S), or $G = \circ F$ and G is ground in T (satisfies lemma). We got that if F is ground in sequent S, then F or $\circ F$ is ground in every sequent in a $\circ$-loop $S \rightsquigarrow S'$.

Case 2) Ground formula $F \in S'$. Suppose, that formula F is not ground on some sequent T inside the $\circ$-loop $S \rightsquigarrow S'$. So, there exist ground formula $G \in T$, that $F \subset_{sf} G$.

There exist ground formula $H \in S$, that $G \subseteq_{sf} H$. According to Lemma 3, $F \subset_{sf} H \in S$ or $F = H$ ($G = \circ H = \circ F$). Since $S \rightsquigarrow S'$ is a $\circ$-loop, $H \in S'$. Formula F is ground in S'. Therefore, $F \not\subset_{sf} H$, and $F = H$.

If $F = H$, then G is ground in T and $G \subseteq_{sf} F (= H)$. So, $G = F$ or $G = \circ F$ and is ground in T. We got that if F is ground in sequent S', then F of $\circ F$ is ground in every sequent in a $\circ$-loop $S \rightsquigarrow S'$.

COROLLARY 1. *If we have derivation tree satisfying the following items:*
– we have rule application with conclusion T and premise T',
– there exist ground formula $\Box F$ (or $\circ\Box F$) in the sequent T,
– formula $\Box F$ or formula $\circ\Box F$ is not a ground in sequent T'.
Then Lemma 5 *says, that sequent T is not inside any $\circ$-loop $S \rightsquigarrow S'$.*

Proof goes straightforward from Lemma 4 and Lemma 5.

In other words, if some ground formula was deleted during some rule application (in bottom-up direction), then we do not need to check any sequent below that rule application in order to catch a loop. The main problem is to identify such a situation, because every time we delete some ground formula, at least one new ground formula appears.

The only rule (in calculus G_3LTL), which may satify above conditions, is $(\to \Box)$, then we take left premise (nonmodal case). So, we can add special indexes for modal operator $\Box$ to catch such a situation.

We add different upper-indexes for every *different* subformula $\Box F$ in the sequent S. The bottom-index will be a set of indexes. The bottom-index is defined according to the following rule: if $\Box^i_U F \subset_{sf} \Box^j_V G$ in sequent S, then $V \subset U$ and $j \in U$.

It means that every ground formula $\Box F$ (or $\circ\Box F$) in any sequent S have empty set as its bottom index: $\Box^i_\emptyset F$ (or $\circ\Box^i_\emptyset F$).

Indexes are some kind of the histories, because they store information abaut applied rules (efficient calculi with used histories may be found in [3,4]).

DEFINITION 13. Sequent calculus for linear temporal logic with inference rules $(\to \vee)$, $(\vee \to)$, $(\to \&)$, $(\& \to)$, $(\to \neg)$, $(\neg \to)$, $(\to \Box^*)$, $(\Box \to)$, $(\circ^*_p)$, with a $\circ$-loop-axiom and with an axiom $\Gamma, \phi \to \Delta, \phi$ we call sequent calculus G_4LTL.

$$\frac{\Gamma^* \overset{\delta}{\to} \Delta^*, A^* \quad \Gamma \to \Delta, \circ\Box^i_U A}{\Gamma \to \Delta, \Box^i_U A}\ (\to \Box^*) \qquad \frac{\Gamma \overset{\circ}{\to} \Delta}{\Sigma, \circ\Gamma \to \Pi, \circ\Delta}\ (\circ^*_p).$$

Here, Σ, Π contains only propositional variables, $\Sigma \cap \Pi = \emptyset$. $\delta \in \{\emptyset, +\}$ and

– if there is no formula $\Box^i_T G$ (or $\circ\Box^i_T G$) $\in \Gamma \cup \Delta$, then $\delta = +$; and Γ^*, Δ^*, A^* are the same as Γ, Δ, A, only index i is fully removed from any subformula,

– if there is some formula $\Box^i_T G$ (or $\circ\Box^i_T G$) $\in \Gamma \cup \Delta$, then $\delta = \emptyset$; and Γ^*, Δ^*, A^* are exactly the same as Γ, Δ, A.

Simple speaking, if we got a sequent marked with $+$, we know, that some ground formula was just deleted and loop cannot appear here. If we delete some ground formula (we also delete some index i), we must get some new ground formula. These new ground formulas will be formulas containing modalized subformula with emptyset as its bottom index.

LEMMA 6. *Sequent S is derivable in sequent calculus G_3LTL if and only if sequent S is derivable in sequent calculus G_4LTL.*

The proof is straightforward from Corollary 1.

So, we get efficient decision procedure for linear temporal logic if we use sequent calculus G_4LTL with restricted loop-check (for both loop types), which:

– checks only sequents marked by $\circ$,

– checks only till the first sequent marked with $+$ (in top-down direction).

3. Conclusion

In this paper, we prove that some restrictions for loop-check for linear temporal logic may be applied without loosing derivability. We prove that any ground formula is modalized and stable in any loop. These restrictions let us to construct sequent calculus with efficient loop-check, because (during loop-check) only several special marked sequents must be checked. The same restriction for ground formulas may by applied for other modal logics (first of all for branching time logic).

References

1. A. Birstunas, Efficient loop-check for KD45 logic, *Lith. Math. J.*, **46**(1), 1–12 (2006).
2. J. Gaintzarain, M. Hermo, P. Lucio, M. Navarro, F. Orejas, A cut-free and invariant-free sequent calculus for PLTL, *Lecture Notes in Computer Science*, **4646**, 481–495 (2007).
3. A. Heuerding, M. Seyfried, and H. Zimmermann, Efficient loop-check for backward proof search in some non-classical propositional logics, in: P. Miglioli, U. Moscato, D. Mundici, M. Ornaghi (Eds.), *Tableaux 96*, *LNCS*, **1071**, 210–225 (1996).
4. M. Mouri, Constructing counter-models for modal logic K4 from refutation trees, *Bull. Section of Logic*, **31**(2), 81–90 (2002).
5. N. Nide and S. Takata, Deducton systems for BDI logic using sequent calculus, in: *Proc. AAMAS'02* (2002), pp. 928–935.

REZIUMĖ

A. Birštunas. Apribojimai ciklų radimui sekvenciniame laiko logikos skaičiavime

Darbe pateiktas sekvencinis skaičiavimas tiesinei laiko logikai, kuris naudoja efektyvų ciklų radimo mechanizmą. Darbe įrodyta, kad atraminės formulės cikluose visada yra modalizuotos ir nekinta. Šie apribojimai leidžia mums ciklų paieškoje apsiriboti keliomis specialiai pažymėtomis sekvencijomis. Analogiški apribojimai gali buti pritaikyti ir kitoms modalumo logikoms (visų pirma skaidaus laiko logikai).

Raktiniai žodžiai: sekvencinis skaičiavimas, laiko logika, efektyvus ciklų radimas.